

UNIVERZA V LJUBLJANI
FAKULTETA ZA RAČUNALNIŠTVO IN INFORMATIKO

Mitja Cukjati

**Sistematični pregled literature o uporabi metodologije
Kanban na področju razvoja programske opreme**

DIPLOMSKO DELO
NA UNIVERZITETNEM ŠTUDIJU

Ljubljana, 2016

UNIVERZA V LJUBLJANI
FAKULTETA ZA RAČUNALNIŠTVO IN INFORMATIKO

Mitja Cukjati

**Sistematični pregled literature o uporabi metodologije
Kanban na področju razvoja programske opreme**

DIPLOMSKO DELO
NA UNIVERZITETNEM ŠTUDIJU

MENTOR: prof. dr. Viljan Mahnič

Ljubljana, 2016

Rezultati diplomskega dela so intelektualna lastnina avtorja. Za objavljane ali izkoriščanje rezultatov diplomskega dela je potrebno pisno soglasje avtorja, Fakultete za računalništvo in informatiko ter mentorja.

Fakulteta za računalništvo in informatiko izdaja naslednjo nalogo:

Tematika naloge:

V skladu s priporočili za pisanje preglednih znanstvenih člankov izdelajte sistematični pregled literature, ki obravnava uporabo metodologije Kanban na področju razvoja programske opreme. Pri tem kot izhodišče uporabite pregledni znanstveni članek, v katerem so zajete študije, objavljene do leta 2012, in ga dopolnite z analizo študij, ki so bile objavljene kasneje. V nalogi najprej opišite načela vitkega razvoja, metodologijo Kanban in pravila za izdelavo sistematičnih pregledov literature. Nato predstavite ugotovitve prej omenjenega članka in na enak način analizirajte študije, objavljene po letu 2012. Iz naloge naj bo razviden postopek iskanja relevantnih študij, njihova klasifikacija, pregled obravnavanih elementov Kanbana ter opredelitev koristi in izzivov pri njegovi uporabi.

IZJAVA O AVTORSTVU DIPLOMSKEGA DELA

Spodaj podpisani Mitja Cukjati sem avtor diplomskega dela z naslovom:

Sistematični pregled literature o uporabi metodologije Kanban na področju razvoja programske opreme

S svojim podpisom zagotavljam, da:

- sem diplomsko delo izdelal samostojno pod mentorstvom prof.dr. Viljana Mahniča,
- so elektronska oblika diplomskega dela, naslov (slov., angl.), povzetek (slov., angl.) ter ključne besede (slov., angl.) identični s tiskano obliko diplomskega dela,
- soglašam z javno objavo elektronske oblike diplomskega dela na svetovnem spletu preko univerzitetnega spletnega arhiva.

V Ljubljani, dne 30. avgusta 2016

Podpis avtorja:

Zahvaljujem se mentorju prof.dr. Viljanu Mahniču za strokovno mentorstvo in dobre nasvete pri izdelavi diplomske naloge. Posebna zahvala gre Ani in staršem za vse spodbudne besede in podporo v času študija in pisanja te naloge. Hvala tudi prijateljem in prijateljicam, ki ste mi stali ob strani, ko je bilo treba.

Mami Milojki in tatu Sergeju.

Kazalo

Povzetek

Abstract

Poglavje 1	Uvod	1
Poglavje 2	Predstavitev pojmov	5
2.1	Metodologija vitkega razvoja	5
2.1.1	Optimizirati celoto (angl. optimize the whole).....	5
2.1.2	Izločanje odpadkov(angl. eliminate waste)	6
2.1.3	Vgraditev kakovosti (angl. build quality in).....	6
2.1.4	Nenehno učenje (angl. learn constantly)	7
2.1.5	Čim hitrejša dostava (angl. deliver fast)	7
2.1.6	Vplesti vsakogar (angl. engage everyone).....	8
2.1.7	Stalno izboljševanje (angl. keep getting better).....	8
2.2	Kanban	8
2.2.1	Lastnosti Kanban	9
2.3	Sistematični pregled literature	22
2.3.1	Uvod	22
2.3.2	Načrtovanje.....	23
2.3.3	Izvedba pregleda.....	27
2.3.4	Poročanje	36
Poglavje 3	Kanban pristop, med agilnostjo in vitkostjo: sistematični pregled	39
3.1	Uvod.....	39
3.2	Določitev problematike in obsega	39
3.3	Metodologija raziskave.....	40
3.3.1	Cilji sistematičnega pregleda literature	41
3.3.2	Iskanje primarnih študij	43
3.3.3	Izbira primarnih študij	44
3.3.4	Postopek pridobivanja podatkov	48

3.3.5	Sinteza podatkov	48
3.3.6	Rezultati	49
Poglavje 4	Izvedba sistematičnega pregleda literature	61
4.1	Uvod	61
4.2	Načrtovanje	61
4.2.1	Protokol pregleda	61
4.2.2	Obrazec za pridobivanje podatkov	65
4.3	Predstavitev rezultatov	65
4.3.1	Rezultati iskanja in ocenjevanja kakovosti literature.....	65
4.3.2	Rezultati raziskave	66
4.3.3	Elementi Kanbana	67
4.3.4	Koristi in izzivi uporabe Kanban pristopa	75
4.3.5	Jasnost obstoječih smernic za vpeljavo Kanban pristopa v organizacijo.....	78
4.3.6	Kanban tabla	80
4.3.7	Povezave med elementi Kanbana in načeli vitkosti.....	80
Poglavje 5	Sklepne ugotovitve	83
Seznam slik		
Seznam tabel		
Literatura		
Priloge		
	Priloga I – Predlog oblike poročila sistematičnega pregleda literature	
	Priloga II – Kodirna shema	
	Priloga III – Seznam virov literature s podrobnostmi izvedbe iskanja	
	Priloga IV – Seznam vključene literature	
	Priloga V – Seznam izključene literature	
	Priloga VI – Predloga obrazca za pridobivanje podatkov	

Seznam uporabljenih kratic

kratica	angleško	slovensko
ARR	Absolute risk reduction	Absolutno znižanje tveganja
CFD	Cumulative flow diagram	Kumulativni diagram poteka
CI	Continuous integration	Zvezna integracija
CRD	Centre for Reviews and Dissemination	Center za preglede in objavave
DARE	Database of Abstracts of Reviews of Effects	
JIT	Just-in-time	Ravno pravočasno
OR	Odds ratio	Relativne kvote
PEH	Perfect Engineering Hours	Dejansko porabljeni čas
POOGI	Proces Of OnGoing Improvment	Proces nenehnega izboljševanja
RR	Risk ratio	Relativno tveganje
SMD	Standardised mean difference	Standardna povprečna razlika
TDD	Test-driven development	Testno voden razvoj
TOC	Theory of constraints	Teorija omejitev
TPS	Toyota Production System	Toyotin sistem proizvodnje
WIP	Work-in-progress	Delo v teku
WMD	Weighted mean difference	Tehtana povprečna razlika

Povzetek

V diplomskem delu je bil izveden sistematični pregled literature, ki obravnava vpeljavo metodologije Kanban na področju razvoja programske opreme. V prvem delu so predstavljeni pojmi metodologije vitkega razvoja, Kanban in postopek sistematičnega pregleda literature. V drugem delu je predstavljen članek *The kanban approach, between agility and leanness: a systematic review*. V članku je predstavljen sistematični pregled literature, ki je bila objavljena med leti 1990 in 2012, na temo vpeljave Kanbana v proces razvoja programske opreme. Cilj sistematičnega pregleda je razjasniti vodila in elemente Kanban pristopa, da bi spodbudili njegovo uspešno vpeljavo na področje razvoja programske opreme z jasnim definiranjem osnovnih pojmov in predstavitev prednosti in slabosti njegove uporabe v IT organizacijah. V tretjem delu je predstavljen sistematični pregled literature, ki je osrednja tema diplomskega dela in temelji na omenjenem članku. Obravnava znanstveno literaturo, ki je bila objavljena po letu 2012. V zadnjem delu so predstavljene ugotovitve sistematičnega pregleda in njihova primerjava z ugotovitvami prej omenjenega članka.

Ključne besede: Kanban, vitek razvoj programske opreme, agilni razvoj programske opreme, sistematični pregled literature

Abstract

The thesis presents a systematic literature review covering the implementation of the Kanban approach within the field of software development. The first section provides background related to the research area by describing lean software development, the Kanban method and the guidelines for undertaking such a review. The second section covers the article *The kanban approach, between agility and leanness: a systematic review* which is a systematic literature review of research studies on the implementation of the Kanban approach in the process of software development, published between 1990 and 2012. The purpose of the article was to illuminate the guiding principles and elements of the Kanban approach in order to increase the likelihood of implementing it successfully by defining the prime elements of Kanban and outlining the advantages and benefits of using it within IT organisations. The results of the systematic literature review are presented in the third section. The analysis was conducted following the guidelines from the first section, covering scientific writings published after 2012. The final section presents the findings of the systematic literature review and their comparison to the findings from the above-mentioned article.

Keywords: Kanban, Lean software development, Agile software development, systematic literature review

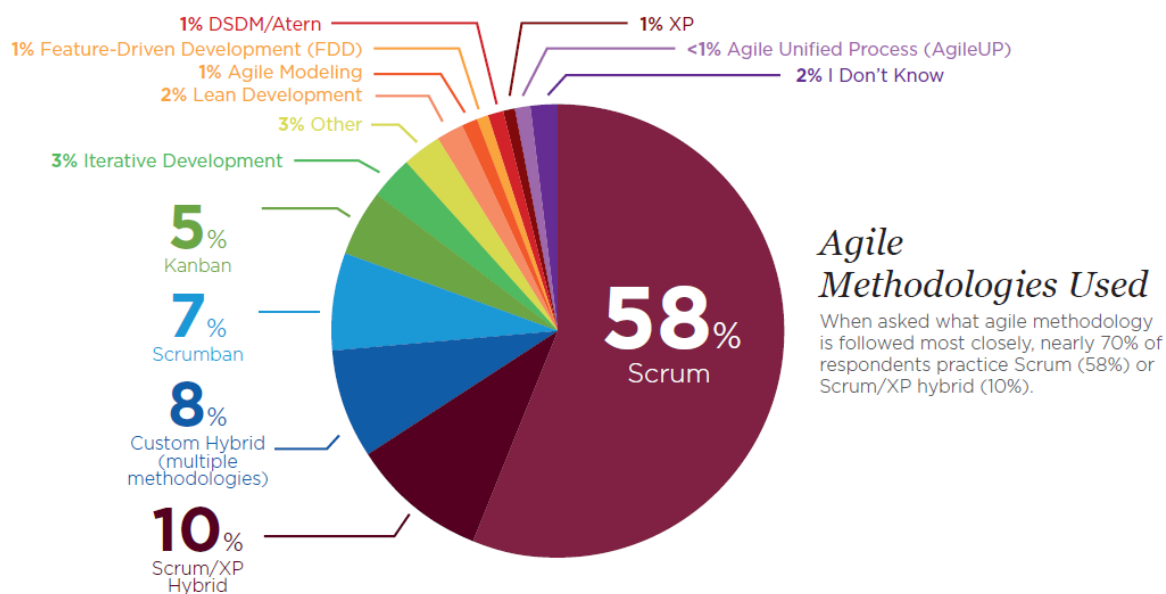
Poglavje 1 Uvod

Začetki kanbana segajo v štirideseta leta dvajsetega stoletja, ko so pri Toyota Motor Company, takrat še razmeroma majhnem akterju v avtomobilski industriji, začeli z uvedbo revolucionarnih idej na področje optimizacije proizvodnega procesa, kot so eliminiranje muda, kaizen in koncept ravno pravočasno (angl. just in time). Muda je japonski izraz za nesmiselnost, neuporabnost, potratnost. Kaizen, prav tako japonski izraz, pa pomeni proces nenehnega izboljševanja. Pri Toyoti so proizvodni sistem poimenovali Toyota Production System, TPS, za snovalca sistema pa veljata Taiichi Ohno in Eiji Toyoda. Ohno je koncept ravno pravočasno v TPS poimenoval kanban, njegova glavna naloga pa je bila usklajevanje materialnega toka v oskrbovalni verigi na dnevni ravni. Beseda kanban v japonščini pomeni signalizacijska kartica, s katerimi se v TPS signalizira potreba po sestavnih delih izdelka. Sistem TPS velja za predhodnika sistemov vitke proizvodnje [1].

Na področju razvoja programske opreme velja kanban za dokaj nov pojem, prvi kanban sistem je bil vpeljan leta 2004 [2]. Med leti 2006 in 2008 je David Anderson začel oblikovati metodo, ki temelji na načelih vitkosti in ob vpeljavi v delovni proces tega nenehno razvija in izboljšuje. Metoda kot orodje za svoje delovanje izrablja kanban sistem, Anderson pa jo je poimenoval Kanban (pozor na veliko začetnico). O razlikovanju kanban sistema in Kanban metode bomo podrobneje pisali v nadaljevanju.

V anketah State of Agile Development Survey, ki jo VersionOne letno opravlja od leta 2006 dalje, se kanban najprej omenja kot orodje Scruma [3]. Kanban kot metodologija, kar sicer ne ustreza Andersonovi definiciji, prvič nastopi leta 2011 [4], v naslednjih letih pa se že kaže nagel porast deleža njegove uporabe z vrhom v letu 2015 s 5%, če se upošteva še hibridno varianto - v kombinaciji s Scrumom, t.i. Scrumban, pa kar z 12% deležem uporabe [5]. Na vrhu lestvice že od prve ankete kraljuje Scrum, njegov delež v letu 2015 znaša 58%, skupaj s hibridnimi različicami pa kar 75%.

AGILE METHODS AND PRACTICES



Slika 1: Stanje uporabe agilnih metodologij na področju razvoja programske opreme v letu 2015 [5].

Namen diplomske naloge je pregledati obstoječo literaturo, ki obravnava vpeljavo Kanbana v proces razvoja programske opreme. V letu 2015 je bil objavljen sistematični pregled literature *The kanban approach, between agility and leanness: a systematic review* [6], v katerem sta si avtorja Al-Baik in Miller zastavila, da bosta razjasnila vodila in elemente Kanbana, s čimer bi spodbudili njegovo uspešno vpeljavo na področje razvoja programske opreme z jasnim definiranjem osnovnih pojmov in predstavitevjo prednosti in slabosti njegove uporabe v IT organizacijah. Obravnavala sta 37 primarnih študij, ki so bile objavljene med leti 1990 in 2012, in v njih odkrila in definirala 20 elementov Kanbana. Odkrila sta vrsto prednosti, ki jih Kanban prinaša v organizacijo, pa tudi ovire na poti do njegove uspešne uvedbe. Večina obravnavanih študij se je osredotočala na implementacijo vitkega razvoja, Kanban pa omenjala kot njegovo najpogostejše uporabljeno orodje. Implementacijo vitke metodologije preko Kanbana navajata kot verjetno najhitreje rastočo metodologijo razvoja na področju programske opreme v zadnjem desetletju. Kljub temu je področje še vedno zelo slabo raziskano. Pomanjkanje standardne definicije Kanbana in njegovih elementov, pomanjkanje podrobnosti in slabo definirane smernice za vpeljavo Kanbana v organizacijo, pomanjkanje smernic za vpeljavo posameznih elementov Kanbana. To so težave, s katerimi se v praksi srečujemo ob vpeljavi Kanbana v organizacijo. Kot pozitivne učinke implementacije Kanbana navajata Kanban tablo, ki se je izkazala kot odlično orodje za vizualizacijo delovnega toka.

Več kot tretjina raziskanih študij pa navaja, da so v organizaciji uspešno uporabili Kanban za upravljanje organizacijskih sprememb in izboljšanje večfunkcijskega ekipnega dela. V zaključku ugotavljata, da je tematika vpeljave Kanbana na področju razvoja programske opreme slabo raziskano, ter da se več kot očitno kaže potreba po sistematičnih študijah o prednostih, ki jih Kanban prinaša v organizacijo, ter o smernicah pri določanju osnovnih elementov Kanbana.

V diplomski nalogi bomo poskušali sistematični pregled v [6] nadaljevati z analiziranjem literature, ki je bila objavljena po letu 2012. Študija v [6] obravnava tudi sivo literaturo, v tem delu pa se bomo omejili le na znanstveno. Da bo postopek pregleda kar se da sistematičen in nepristranski, bomo pri tem uporabili enako metodologijo raziskave, kot sta jo uporabila avtorja v [6]. Metodologija temelji na smernicah, ki jih Kitchenham in Charters podajata v *Guidelines for performing Systematic Literature Reviews in Software Engineering* [7].

Diplomsko delo je sestavljeno iz štirih delov. V prvem delu (0) so predstavljeni ključni pojmi: metodologija vitkega razvoja, Kanban, ter postopek sistematičnega pregleda literature. V drugem delu (0) je podrobneje predstavljen članek *The kanban approach, between agility and leanness: a systematic review*. V tretjem delu (0) je predstavljen postopek in rezultati izvedenega pregleda literature. V zadnjem delu (0) so predstavljene ugotovitve in primerjava z ugotovitvami raziskave *The kanban approach, between agility and leanness: a systematic review*.

Poglavje 2 Predstavitev pojmov

2.1 Metodologija vitkega razvoja

Vitka proizvodnja (angl. lean production) se je prvič pojavila v poznih štiridesetih letih dvajsetega stoletja v avtomobilski industriji v japonski tovarni Toyota Motor Company z uvedbo proizvodnega sistema Toyota Production System (TPS). Sistem sta zasnovala Taiichi Ohno in Eiji Toyoda. Ohno je v TPS vpeljal tedaj revolucionarne principe in orodja, kot so zmanjšanje odpadkov (angl. waste), proces stalnega izboljševanja, tehniko petih zakajev pri odpravljanju napak (odpravljanje izvora napake) in povsem nov način usklajevanja materialnega toka v oskrbovalni verigi na dnevni ravni, t.i. koncept "ravno pravočasno" (angl. just-in-time, JIT), ki so ga pri Toyoti poimenovali kanban [1].

Leta 1996 sta Womack in Jones na podlagi TPS določila pet načel vitkega razmišljanja: določitev vrednost z vidika kupca (angl. specify value), opredelitev načrta vrednostnega toka (angl. identify the value stream), vzpostavitev toka z izločanjem odpadkov (angl. make the value-creating steps flow), izdelava na zahtevo (angl. pull), in stremljenje k popolnosti (angl. perfection) [8].

Prvi zapisi v strokovni literaturi o uvajanju konceptov vitkosti na področje razvoja programske opreme so se pojavili že v zgodnjih devetdesetih letih dvajsetega stoletja, slabo desetletje pred nastankom manifesta agilnosti leta 2001 [9]. Tedanji glavni cilj vitkega pristopa je bil izločanje odpadkov iz procesa razvoja programske opreme [10] [11].

Leta 2003 sta Mary in Tom Poppendieck na področje razvoja programske opreme vpeljala 7 načel vitkosti in množico orodij razmišljanja za prenos teh načel v uporabo v okviru agilnih metodologij [12]. Ta načela so kratko predstavljena v nadaljevanju.

2.1.1 Optimizirati celoto (angl. optimize the whole)

Na organizacijo lahko gledamo kot na sistem. Sistema pa ne tvorijo le njegovi sestavni deli kot skupek neodvisnih podsistemov, ampak njegov obstoj omogoča tudi njihovo medsebojno delovanje. Sistem, ki je sestavljen iz najboljših komponent ni nujno najboljši. Na kakovost sistema kot celote torej vpliva tudi mera kakovosti medsebojnega sodelovanja med njegovimi sestavnimi deli.

Pri sistemskem obravnavanju organizacije se pojavljajo trije vzorci: doseganje mej rasti (angl. limits to growth), izmikanje odgovornosti (angl. shifting the burden) in optimiziranje

sestavnih delov sistema (angl. suboptimization). Pri prvem vzorcu velja, da se kljub doseganju zelenih rezultatov procesa pojavljajo stranski učinki, ki negativno izenačujejo dosežene rezultate. Razlog je v (navadno rigoroznih) ukrepih, s katerimi smo rezultate procesa dosegli. Namesto izvajanja pritiskov za doseganje čim boljših rezultatov, je boljše, da odkrijemo ovire, ki doseganje rezultatov zavirajo, in jih odpravimo.

Izmikanje odgovornosti se pojavi, ko se naleti na problem ali napako, ki je ni možno prezreti oz. zanemariti. Pogosto se v takem primeru raje odpravlja simptome težave, kot pa išče njen izvor. Vitki pristop predlaga uporabo orodja petih zakajev (angl. five whys), s katerim se postopoma prebijemo do izvora težave.

Pri optimiziranju sistema je treba nanj vedno gledati kot na celoto. Bolj kot je sistem zapleten, raje ga delimo na manjše dele, ki jih nato posamično izboljšujemo. Ni pa nujno, da z optimiziranjem podsistemov izboljšujemo tudi sistem kot celoto.

2.1.2 Izločanje odpadkov(angl. eliminate waste)

Pod pojmom odpadek, razumemo vse kar za kupca ne predstavlja neposredne dodane vrednosti ali vsa prizadevanja, ki ne doprinesejo k znanju o tem, kako dodano vrednost ustvariti na bolj učinkovit način. Izločanje odpadkov je temeljno načelo vitkega razvoja iz katerega izhajajo vsa ostala načela. Tako je prvi korak k vpeljavi vitkosti v organizacijo prav odkrivanje in izločanje odpadkov. Al-Baik in Miller sta sedem vrst odpadkov s področja proizvodnje pretvorila v sedem vrst odpadkov, ki so navzoči na področju razvoja programske opreme: delno opravljeno (nedokončano) delo, odvečni procesi, odvečne funkcionalnosti, preklapljanje med opravili, čakanje (brezdelje), nepotrebni premiki, in napake.

Kot dober način za odkrivanje odpadkov avtorja navajata diagram vrednostnega toka, to je shema korakov v procesu organizacije, ki privedejo kupčevo zahtevo do uresničitve – izdelka oz. storitve.

2.1.3 Vgraditev kakovosti (angl. build quality in)

Shingo je v [13] izvajanje nadzora v kontekstu odkrivanja in odpravljanja napak delil v dve vrsti: raziskovanje vzroka po odkritju napak in izvajanje nadzornih ukrepov v izogib nastajanju napak. Za doseganje kakovosti je pomembno, da težimo k slednjemu, torej preventivnemu ukrepanju. V primeru nastanka napak pa vitki razvoj spodbuja takojšnje odpravljanje napake in izogibanje kopičenju in odlašanju odpravljanja napak.

Za uporabni orodji pri zagotavljanju kakovosti se navajata testno voden razvoj (angl. test-driven development, TDD) in postopek zvezne integracije (angl. continuous integration, CI), ki pripomore k zgodnjemu odkrivanju napak v izdelku.

2.1.4 Nenehno učenje (angl. learn constantly)

Dejstvo je, da pridemo med razvojem izdelka do vedno novih spoznanj. Znanje o izdelku in njegovem kontekstu se tako povečuje. Vitki razvoj predlaga dva pristopa kako novo nastalo znanje vnovčiti v nadaljnjem razvoju izdelka.

Prvi pristop imenujemo pristop s predhodnim učenjem (angl. learn first). Po tem pristopu za odločitve, ki so nespremenljive oz. nas kasnejše spremembe lahko drago stanejo, najprej raziščemo različne možnosti – se o problematiki najprej podučimo. Takšne odločitve so na primer o arhitekturi sistema, izbiri programskega jezika in orodij, ipd. Pomembno je, da se s temi odločitvami odlaša čim dlje, kar nam omogoča, da ohranjamo možnosti odprte, medtem ko naše znanje o tematiki raste.

Drugi pristop se imenuje nenehno učenje (angl. learn constantly). Pristop je inkrementalen. Začnemo z izgradnjo za kupca najmanjšega sprejemljivega števila funkcionalnosti. Izdelek nato s pogostimi izdajami verzij dopolnjujemo, kupčevo povratno informacijo pa izkoriščamo za krepitev znanja o izdelku, kar nam olajša sprejemanje odločitev med razvojem. S tem pristopom omejimo možnosti za razvoj funkcionalnosti, ki uporabniku niso potrebne. Mimogrede, take funkcionalnosti v vitkem razvoju obravnavamo kot odpadke.

Seveda lahko uberemo tudi srednjo pot. Pri tem pa moramo upoštevati, da je treba osvojeno znanje - odkrite možnosti in omejitve - upoštevati pri sprejemanju odločitev, hkrati pa ne zanemariti dejstva, da se vsebina sistema v razvoju s časom nenehno spreminja.

2.1.5 Čim hitrejša dostava (angl. deliver fast)

Prednost hitre dostave z vidika kupca je povečanje fleksibilnosti kupčevega poslovanja, ki jo nudijo nove verzije programske opreme. Na strani razvijalca programske opreme pa se kaže v večji razpoložljivosti zaposlenih (naloge so manj obsežne in hitreje zaključene), manjšem številu delno narejenih oz. nezaključenih opravil in manjšem številu nakopičenih neodpravljenih napak. Dlje kot z izdajo izdelka odlašamo, oz. bolj kot je izdaja obsežna, večja je verjetnost za pojavljanje napak.

Nenazadnje lahko s hitro dostavo dosežemo, da kupcu izdelek dostavimo preden si lahko o nakupu premisli in izdelek kupi pri našem tekmeču.

2.1.6 Vplesti vsakogar (angl. engage everyone)

Za uspešno vzpostavitev vitkega razvoja je bistvenega pomena, da se ljudem odobri ustrezna pooblastila, spodbuja skupinsko delo in duh, ter prenese sprejemanje odločitev na najnižjo možno raven organizacije.

Na proces razvoj programske opreme je treba gledati širše od samega razvoja, vanj navadno ne bodo vpleteni le razvijalci programske opreme. Proces navadno vključuje tudi ljudi z različnih področij, kot so odnosi s strankami, oblikovalstva, razvoja programske opreme, testiranja, podpore uporabnikom in tudi financ.

2.1.7 Stalno izboljševanje (angl. keep getting better)

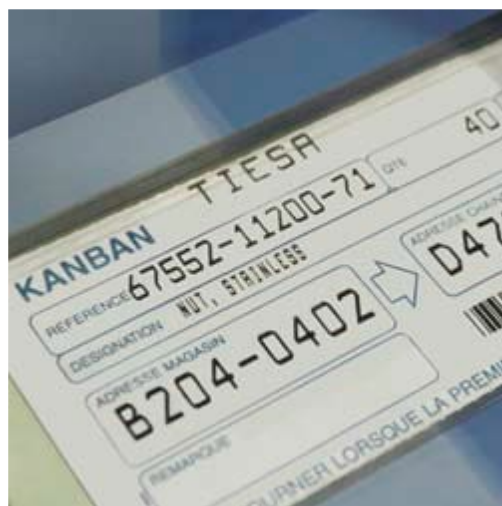
Pri vitkem razmišljanju velja razmišljanje, da so za dani problem redko najboljša rešitev že uveljavljeni postopki, ne glede na to kako dobro delujejo v drugih situacijah. Po tem načelu neka absolutno najboljša praksa, metodologija, oz. način dela, ne obstaja. Vitko razmišljanje priporoča, da organizacije, ki že uporabljajo metodologije, kot so XP in Scrum, le te obravnavajo kot začetno točko, ki jih je treba prilagajati in izpopolnjevati.

2.2 Kanban

Naj že kar na začetku razjasnimo pomen in uporabo besede kanban. Gre za besedo v japonskem jeziku, ki v dobesečnem prevodu pomeni signalizacijska tabla.

V tem delu bomo besedo kanban uporabili za pojmovanje dveh različnih stvari: kanban sistem in Kanban (pozor na veliko začetnico) metodo.

Kanban sistem spada v družino izvlečnih sistemov (pull system). Bistvo teh sistemov je, da se delo začne opravljati takrat, ko so na voljo potrebne kapacitete za njegovo opravljanje. Novo delo je povlečeno v sistem. Alternativa izvlečnih sistemov so potisni sistemi (angl. push system), pri katerih se novo delo glede na zahteve potisne v sistem. Kot smo že omenili, je bil kanban sistem prvič vpeljan v TPS, kot orodje za upravljanje oskrbovalne verige po principu ravno pravočasno, JIT, za njegov nastanek pa je zaslužen Taiiji Ohno.



Slika 2: Primer kanban kartice v sistemu TPS [8].

Kanban sistem deluje tako, da je delovni proces razdeljen v korake. Ko se v nekem koraku pojavi zahteva po učinku dela, se ta sporoči (signalizira) predhodnemu koraku v procesu. V TPS so zahtevo signalizirali s kanban kartico. Delo se lahko v nekem koraku začne le ob prejetju zahteve iz naslednjega koraka delovnega procesa. V obtoku je določeno (dogovorjeno) število kanban kartic, ki hkrati pomeni kapaciteto sistema – omejitev dela v teku (angl. WIP limit). Vsaka kartica se lahko dodeli natanko enemu predmetu dela, s katerim se pomika skozi delovni proces. Ko se delo, kateremu je dodeljena kartica zaključi, se kartica sprosti in se lahko dodeli naslednjemu predmetu dela.

Prvi kanban sistem na področju razvoja programske opreme je bil vpeljan leta 2004 v podjetju Microsoft [2].

David Anderson je med leti 2006 in 2008 začel oblikovati metodo, ki temelji na načelih vitkosti in ob vpeljavi v delovni proces tega nenehno razvija in izboljšuje. Metoda kot orodje za svoje delovanje izrablja kanban sistem, Anderson pa jo je poimenoval Kanban. Ko besedo Kanban zapišemo z veliko začetnico, torej mislimo na Kanban kot metodo [2].

Anderson v [2] poudarja, da Kanban ni (samostojna) metodologija razvoja programske opreme, ali projektnega vodenja. Kanban je metoda, ki zahteva, da je pred njeno implementacijo v organizaciji že v uporabi neka metodologija, ki jo Kanban nato korakoma izboljšuje in izpopolnjuje s procesom nenehnih izboljšav.

2.2.1 Lastnosti Kanban

Kanban v organizacijo uvaja vitko delovanje z uporabo petih temeljnih lastnosti [2]:

- Vizualizacija delovnega toka (angl. Visualize Workflow),
- Omejitev obsega dela v teku (angl. Limit Work-in-Progress),
- Merjenje in upravljanje delovnega toka (angl. Measure and Manage Flow),
- Jasna pravila delovnega procesa (angl. Make Process Policies Explicit),
- Uporaba modelov za odkrivanje priložnosti za izboljšave (angl. Use Models to Recognize Improvement Opportunities).

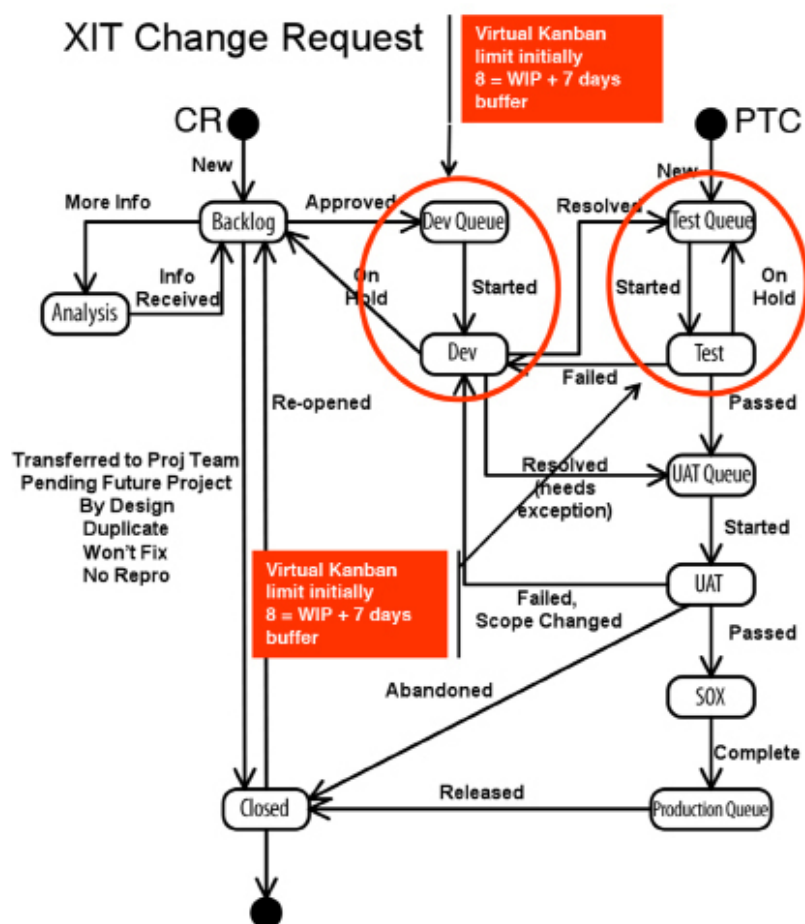
2.2.1.1 Vizualizacija delovnega toka

Pri vizualizaciji delovnega toka poskušamo z analizo odkriti spremembe stanj v njem. Te se navadno kažejo kot spremembe aktivnosti, ki tvorijo nove informacije o predmetu dela, in

prehodi med njimi. Za primer lahko vzamemo aktivnost analiza zahtev. Ob njenem izvajanju na predmetu dela nastajajo informacije. Na neki točki izvajanja te aktivnosti začne obseg nastalih informacij upadati. Ob tem trenutku navadno rečemo, da je bila analiza zahtev opravljena in je predmet dela v stanju "analiziran". Na predmetu dela zato začnemo izvajati naslednjo aktivnost, npr. načrtovanje, ki začne o njem tvoriti spet nove informacije.

Cilj vizualizacije delovnega toka je pridobiti čim več podatkov in znanja o obstoječem delovnem toku. Šele ko dobro razumemo trenutno stanje, lahko z ustreznimi ukrepi poskušamo delovni tok izboljšati.

Prvi korak pri vizualizaciji delovnega toka je izdelava skice ali modela delovnega toka. Te lahko izdelamo z diagramom stanj (angl. state chart), zadostuje pa že preprosta skica, diagram poteka, ipd. [2] [14].



Slika 3: Primer vizualizacije delovnega toka z diagramom stanj [2].

Kanban tabla

Kanban tabla je tipično orodje, ki v Kanbanu služi za prikaz delovnega toka in koordinacijo.

Stolpci na Kanban tabli predstavljajo aktivnosti delovnega toka in po potrebi čakalne vrste, ki predstavljajo stanje v katerem se delovna naloga nahaja (npr. pripravljen za testiranje). Stolpci si sledijo od leve proti desni v enakem vrstnem redu, kot se pojavljajo znotraj delovnega toka. Pri začetni zasnovi stolpcev na tabli imamo načeloma proste roke, navadno pa so to aktivnosti in stanja, ki smo jih odkrili z analizo obstoječega delovnega toka ob vpeljavi Kanban, lahko pa izberemo tudi drugačen pristop, kot je na primer izbira aktivnosti slapovnega modela razvoja ali preprost seznam aktivnosti *narediti*, *v delu* in *dokončani* (angl. to do, in progress, done). Aktivnosti se navadno razdeli v "podstolpca" v *delu* in *zaključeni*. Skrajno levo dodamo še stolpec Seznam zahtev (angl. input queue), ki predstavlja vhodni seznam delovnih nalog. Delovni tok teče od leve strani table proti desni. Navadno se prioritete delovnih nalog nakazuje s postavitvijo delovne naloge na tabli – delovne naloge z višjo prioriteto na tabli so postavljene višje od tistih z nižjo prioriteto.

Tablo lahko po potrebi razdelimo tudi v vrstice glede na vrsto delovne naloge, na primer nova zahteva, zahteva za popravek v produkciji, napaka, ipd.

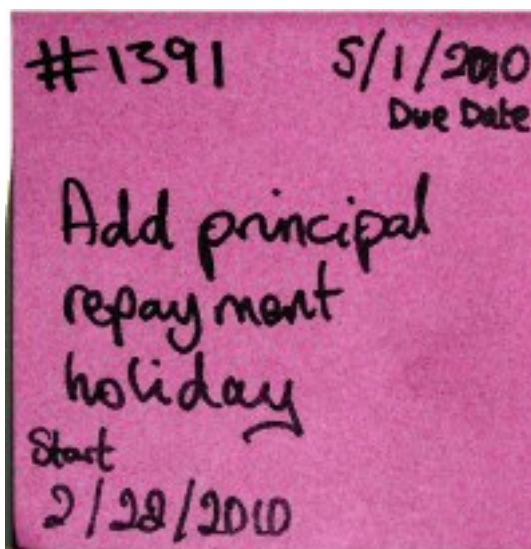


Slika 4: Primer Kanban table [2].

Omejitve dela v teku (angl. WIP limit) se navaja nad vsakim posameznim stolpcem ali nad skupino združenih stolpcev. Če je število delovnih nalog, ki se nahaja v stolpcu manjše od omejitve dela v teku v tem stolpcu, je to znak, da se lahko v stolpec "povleče" (angl. pull) novo delovno nalogo iz stolpca na levi. Pri tem je treba poudariti, da število delovnih nalog v stolpcu ne sme nikoli presegati omejitve dela v teku [2].

Prikaz delovne naloge na Kanban tabli

Vsaka delovna naloga je na tabli predstavljena na svoji kartici. Na kartico se lahko zapiše različne informacije o delovni nalogi. Informacije, ki so na kartici prikazane, je treba izbrati tako, da olajšajo delovanje kanban sistema, olajšajo odločanje pri vodenju projekta, spodbujajo posameznike k samostojnemu izbiranju novih delovnih nalog za izvajanje, ter povečajo preglednost poteka procesa dela, ciljev in tveganj projekta. Narava Kanbana je, da posameznike spodbuja k samostojnemu odločanju, razporejanju svojega dela, upoštevanju prioritet, ter izkazovanju medsebojnega spoštovanja in krepitvi zaupanja v sistem kot tak. Tudi zasnova kartice ima pri tem pomembno vlogo. Za primer lahko naštejemo nekaj pojmov, ki se jih pogosto navaja na karticah: identifikator naloge v elektronskem sistemu za sledljivost delovnih nalog, naslov naloge, kratek opis, datum nastanka, rok za dokončanje in ime osebe, ki nalogo izvaja. Navadno se ime osebe označuje izven kartice (na tabli) ali na tak način, da se lahko enostavno zamenja (npr. uporaba nalepk, magnetkov ipd.), saj bo naloga med potovanjem po Kanban tabli zamenjala več izvajalcev. Pogosto se za označevanja izvajalca naloge uporablja t.i. avatar – uporabnikova slika ali ikona. Na karticah se lahko uporablja razne znake, ki imajo za člane ekipe določen pomen, npr. z zvezdico lahko označimo naloge, ki presegajo potrebni čas (angl. lead time) [2].



Slika 5: Primer prikaza delovne naloge s fiksnim rokom izvedbe na post-it listku [2].

2.2.1.2 Omejitev obsega dela v teku

Omejitev obsega dela v teku pove koliko nalog je lahko (v posameznem stolpcu ali skupini stolpcev Kanban table) hkrati v izvajanju. Na Kanban tabli so omejitve navadno označene nad stolpci ali skupinami stolpcev.

Odločitev o omejitvah obsega dela v teku je ob vpeljavi Kanbana ena najpomembnejših. Omejitve je treba sprejeti soglasno z zaposlenimi na različnih slojih organizacije. S tem se doseže disciplina pri spoštovanju omejitev, hkrati pa se odgovornost za odločitev porazdeli. Pravila za postavitve omejitev ni, obstajajo pa smernice, ki se ločijo glede na vrsto koraka v delovnem toku – delovna aktivnost, čakalne vrste (angl. queues), izravnalne vrste (angl. buffer bottlenecks) in vhodni seznam zahtev (angl. input queue). Omejitve obsega dela v teku se ne določajo enkratno, ampak se s pridobivanjem informacij o učinkovitosti delovanja sistema po njihovi vpeljavi postopno prilagajajo. Ločimo tri načine dela, ki se razvijejo kot posledica vpeljanih omejitev: sodelovalen, samostojen in večopravilen (angl. multitask). Če določimo omejitev enako številu ljudi, ki bodo predvidoma izvajali delovne naloge na neki aktivnosti delovnega toka, bomo spodbujali samostojen način dela. Če omejitev določimo večjo od števila ljudi, bomo s tem spodbujali večopravilnost – posamezen razvijalec bo lahko hkrati delal na več nalogah. Če določimo manjšo omejitev od števila ljudi, bomo spodbujali delo v skupinah, saj bo več ljudi hkrati izvajalo isto nalogo [2].

V zvezi z omejitvami za delovne aktivnosti avtor v [2] navaja, da se na podlagi empiričnih raziskav kot optimalno omejitev podaja dve nalogi na enega človeka. Po drugi strani pa pravi,

da formula za izračun optimalnega števila za omejitve ne obstaja. Priporoča vpeljavo najbolj smiselne omejitve, spremljanje rezultatov vpeljave in po potrebi ponovno prilagoditev.

Čakalne vrste se uporabljajo na prehodih med aktivnostmi delovnega toka. Ko je delo na nalogi v aktivnosti zaključeno, se lahko nalogo postavi v čakalno vrsto, kjer čaka na izvajanje v naslednji aktivnosti. Navadno do potrebe po čakalnih vrstah pride, če so omejitve obsega dela v teku prestrogo določene (ena naloga na osebo, par ali majhno ekipo). Z njimi izravnamo nihanja v delovnem toku in preprečujemo zaustavitve toka – ohranjamo enakomernost delovnega toka. Čakalne vrste naj bojo čim manjše. Če v delovnem toku prihaja do zaustavitev in ostajajo ljudje brez dela, je treba omejitve na čakalnih vrstah povečati. Če pa smo na aktivnosti določili omejitve, ki je večja od ene naloge na osebo, smo s tem čakalno vrsto že vključili v izvedbo aktivnosti. V takem primeru lahko stolpec z aktivnostjo razdelimo v dva: v delu in končani.

Izravnalne vrste uporabljamo za odpravljanje ozkih grl v delovnem toku (angl. bottlenecks). Tudi na izravnalnih vrstah naj bodo omejitve čim manjše. Ključ pri določanju omejitev je, da morajo biti te dovolj velike, da zagotovimo konstanten delovni tok in se hkrati ognemo brezdelju (angl. idle time) na mestih, ki predstavljajo ozko grlo sistema.

Vpeljava čakalnih in izravnalnih vrst ima za posledico podaljšanje potrebnega časa (angl. lead time). Pozitivna stran tega je, da z njimi ohranjamo konstanten delovni tok, in tako povečamo predvidljivost dolžine potrebnega časa, pa tudi prepustnost sistema se s tem poveča.

Na velikost omejitve vhodnega seznama zahtev vpliva pogostost izvajanja aktivnosti za določanje prioritet nalog in prepustnost sistema. Tudi na tem mestu je treba poudariti, da se začetno postavljeno omejitve prilagaja v skladu z empirično pridobljenimi podatki med izvajanjem delovnega toka.

Izjemoma lahko kateremu od korakov delovnega procesa ne določimo omejitve obsega dela v teku. To lahko storimo, ko smo prepričani, da so resursi v tem koraku neskončni in vedno na voljo.

2.2.1.3 Merjenje in upravljanje delovnega toka

Spremljati je treba tok delovnih nalog skozi vsak korak delovnega toka. Pozorni moramo biti na hitrost toka, in da je tok enakomeren, tekoč. Stremimo k hitremu in gladkemu toku brez prekinitev, kar se kaže v hitri odzivnosti na zahteve, zmanjšani možnosti za nastanek tveganj, izogibanju nastajanja stroškov zaradi zamud, ter povečanju predvidljivosti delovanja [14].

Sistem, ki uporablja Kanban, mora biti predvidljiv, delovati kot je bilo načrtovano, poudarek mora biti na konstantnem toku in na nenehnem izboljševanju sistema.

Avtor v [2] predlaga spremljanje naslednjih kazalnikov: spremljanje WIP, potrebni čas, doseganje roka, prepustnost, težave in naloge, ki preprečujejo končanje drugih nalog, učinkovitost toka in napake.

Za spremljanje WIP mej lahko uporabimo kumulativni diagram delovnega toka, na diagramu pa prikažemo število prispelih zahtev (delovnih nalog), WIP in zaključenih zahtev v odvisnosti od časa. Da kanban sistem deluje ustrezno, se na grafu kaže tako, da so pasovi enakomerni, in da v vrednostih ni poskokov.

S spremljanjem potrebnega časa lahko nadzorujemo predvidljivost delovanja procesa razvoja, ter zanesljivost napovedovanja izdaj izdelkov. Odgovorimo lahko na vprašanje kako hitro smo delovno nalogo pripeljali od pojava uporabnikove zahteve do namestitve v produkcijskem okolju, in ali smo nalogo opravili v zastavljenem potrebnem času? Poleg potrebnega časa lahko spremljamo še povprečni potrebni časa, ki je uporaben predvsem za spremljanje delovanja v splošnem.

Pri merjenju doseganja roka odgovarjamo na vprašanje: ali je bila uporabnikova zahteva izdelana pravočasno. Preverjamo torej razliko med ocenjenim potrebnim in dejanskim potrebnim časom. Tako lahko odkrijemo ali sistem deluje predvidljivo. Ko je rok za izdelavo fiksno določen, je seveda najpomembnejši podatek, ali je bila naloga zaključena pravočasno.

Prepustnost sistema (angl. throughput) se meri v številu delovnih nalog, ki so bile zaključene (so sistem zapustile) v določenem obdobju (npr. enem mesecu). Cilj je, da prepustnost našega sistema nenehno povečujemo. Za začetek je dovolj dobro, da prepustnost merimo v številu opravljenih nalog. Ko se organizacija dovolj razvije, pa lahko preidemo na druge mere, kot so število točk uporabniških zgodb (angl. story points), funkcijskih točk, ipd. V končni fazi jo lahko ovrednotimo tudi zneskovno (npr. v Eur / mesec). Prepustnost se v Kanban uporablja kot indikator kakovosti delovanja organizacije in za dokazovanje nenehnega izboljševanja delovanja organizacije.

S spremljanjem težav in nalog, ki preprečujejo končanje drugih nalog, odkrivamo kako dobro se organizacija odziva nanje, ter njihov vpliv na delovanje organizacije. Če se kažejo težave pri doseganju roka za izvedbo, se te odražajo tudi na grafu težav in nalog, ki preprečujejo končanje drugih nalog.

Učinkovitost toka merimo kot razmerje med potrebnim časom naloge in časom izvajanja, t.j. čas dejanskega izvajanja dela na nalogi. Učinkovitost toka nam pokaže koliko manevrnega prostora za izboljšave še imamo.

Napake predstavljajo oportunitetni strošek in vplivajo na potrebni čas in prepustnost Kanban sistema. Zanima nas predvsem število napak v primerjavi s skupnim WIP in prepustnostjo, ki naj bo s časom čim bližje 0 [2].

2.2.1.4 Jasna pravila delovnega procesa

Delovni proces določa množica pravil, po katerih se udeleženi v procesu vedejo. Navadno so ta pravila določena s strani vodstva organizacije in so načrtovana tako, da obvladajo tveganja in kupcem zagotavljajo željeni nivo kakovosti izdelkov oz. storitev.

Izboljšanje delovnega procesa je mogoče šele, ko je ta dobro in jasno opredeljen. Če to ni tako, so interpretacije delovnega procesa običajno čustveno obarvane, anekdotične in pristranske. Z jasno opredelitvijo lahko dosežemo bolj racionalno, empirično in nepristransko obravnavo, ki omogoča lažje doseganje ciljev pri izboljšanju procesa.

S kombinacijo jasnih pravil, transparentnosti v delovanju, ter vizualizacijo delovnega procesa dosežemo spodbujanje posameznikov k samostojnemu sprejemanju odločitev in odgovornosti [2].

2.2.1.5 Uporaba modelov za odkrivanje priložnosti za izboljšave

Za odkrivanje priložnosti za izboljšave in uvajanje izboljšav je treba ubrati znanstveni pristop, ki pomeni nepristranski, predvidljiv, ponovljiv in predvsem nadzorovan postopek. Znanstveni pristop pomeni, da se z implementiranjem ustreznega modela najprej odkrije možnosti za izboljšave. Po vpeljanih spremembah se lahko rezultat opazuje z opravljanjem meritev kazalcev. Dejanski učinek spremembe pa se lahko primerja s predvidenim, ter oceni, ali smo z njim dosegli želeno izboljšavo ali ne [14].

Obstaja več modelov za odkrivanje priložnosti za izboljšave. Avtor v [2] predstavi in predlaga uporabo naslednjih: pet korakov osredotočanja (angl. five focusing steps) iz področja teorije omejitev (angl. theory of constraints, TOC), množico načel vitkega razmišljanja, ki odvečne aktivnosti opredeljuje kot (ekonomske) stroške, ter nekatere različice, ki se osredotočajo na spoznavanje in zmanjševanje spremenljivosti.

Teorija omejitev uporablja pet korakov osredotočanja kot glavno orodje stalnega izboljševanja, ki ga imenuje POOGI (angl. Process Of OnGoing Improvment). Koraki so naslednji:

- Prepoznavaj omejitve. V tem koraku se osredotočamo na odkrivanje ozkih grl našega vrednostnega toka.
- Izkoristi zmožnosti omejitve. V tem koraku odkrivamo kakšna je največja možna prepustnost ozkega grla, ter jo primerjamo z dejansko. Vprašamo se na kakšen način bi lahko dosegli najboljše delovanje ozkega grla, ter kaj je treba spremeniti, da takšno delovanje dosežemo.
- Podredi delovanje celotnega sistema spremembam, ki si jih odkril v prejšnjem koraku. Za povečanje prepustnosti ozkega grla je morda potrebno narediti spremembe na različnih delih vrednostnega toka, delovanje le tega je treba spremembam podrediti.
- Osredotoči se na omejitve. V tem koraku se predvideva, da kljub temu, da smo s spremembami iz drugega koraka dosegli največjo možno prepustnost ozkega grla, prepustnost še vedno ne dosega željenega nivoja. V četrtem koraku je treba vpeljati izboljšave, ki povečujejo kapaciteto in prepustnost ozkega grla do te mere, da ga odpravijo. Ozko grlo tako postane nek drugi del sistema.
- Izogibaj se stagnaciji. Poišči novo omejitev in se vrni na drugi korak. Ta korak pravi, da je treba po vpeljavi izboljšav počakati, da se sistem umiri. Za tem pa je treba korake ponovno ponoviti. Tako se stimulira stalno izboljševanje.

Vitko razmišljanje pojmuje aktivnosti delovnega toka, ki ne doprinašajo dodane vrednosti končnemu izdelku, kot odpadke. Obstajajo pa tudi take aktivnosti, ki le posredno doprinašajo dodano vrednost, in bi bile zato po načelih vitkosti opredeljene kot odpadki (npr. dnevni sestanki). Zato je dobro, da vseh odpadkov ne mečemo v isti koš. Anderson v [2] odpadke najprej pojmuje z ekonomskega vidika kot stroške. Nato jih razdeli v tri skupine: stroški transakcije, stroški usklajevanja in breme napak (angl. failure load).

Stroški transakcije so stroški, ki nastanejo ob vzpostavitvi (angl. setup costs) in stroški, ki nastanejo po opravljeni (angl. cleanup costs) transakciji. Pojem transakcija razumemo široko – lahko gre za projekt, iteracijo razvoja, scrum šprint, ali posamezno aktivnost. Primeri vzpostavitev transakcijskih stroškov iteracije razvoja so planiranje, izbira delovnih nalog, ki se bodo v iteraciji izvedle, ocenjevanje časovne zahtevnosti nalog, finančno načrtovanje, razporejanje kadrov, postavitve razvojnega okolja, ipd. Primeri zaključnih transakcijskih stroškov so integracije izdelkov iteracije, izdaja verzije (dostava izdelka), aktivnosti retrospektive in odstranitev orodij razvojnega okolja.

Ko dva ali več ljudi dela z namenom doseganja istega cilja, je med njimi potrebno usklajevanje. Stroški usklajevanja so tiste aktivnosti, v katerih izvajamo komunikacijo in razporejanja med člani ekipe. Te aktivnosti neposredno ne doprinašajo dodane vrednosti za kupca, vendar so informacije, ki nastanejo z njihovim izvajanjem, bistvenega vpliva na končni izdelek in so zato nepogrešljive. Takih aktivnosti ne skušamo vedno odpravljati, ampak želimo njihov obseg zmanjšati do te mere, da je izvajanje še koristno.

Breme napak se kaže v uporabnikovih zahtevah, ki bi se jim lahko ognili, če bi mu v preteklosti izdelali bolj kakovostni izdelek. Primer bremena napak so klici v klicni center za podporo uporabnikom (angl. help desk) – če je izdelek ustrezne kakovosti, je zahtev za podporo uporabnikov manj. Pod breme napak spadajo tudi popravki napak v produkciji, ki so nefunkcionalne narave – ne vplivajo na funkcionalnosti sistema, ampak njegovo uporabo olajšajo oz. izboljšajo (npr. razdelitev zaslonske maske v dve ločeni zaslonski maski zaradi preglednosti, ipd.). Breme napak sicer še vedno doprinaša neko dodano vrednost izdelku, vendar gre v tem primeru za dodano vrednost, ki bi morala biti ustvarjena že od vsega začetka. Zmanjšanje bremena napak za organizacijo pomeni: zmanjšanje oportunitetnih stroškov zaradi pozne izdaje (zamud), ostanek energije, ki je lahko vložena v izdelavo dodatnih funkcionalnosti, v nekaterih primerih pa celo zmanjšanje števila zaposlenih in tako znižanje stroškov dela.

Anderson v [2] navaja preprosto pravilo za ugotavljanje, ali aktivnost delovnega toka spada med odpadke, ali ne. Za aktivnost, ki jo analiziramo, se vprašamo, ali bi aktivnosti povečali obseg? V primeru, da aktivnost ni odpadek, bi povečanje obsega povečalo tudi doprinos k dodani vrednosti za kupca. Če je odgovor na vprašanje ne, potem gre najverjetneje za odpadek in je treba obseg minimizirati oz. aktivnost odstraniti iz delovnega toka.

Če želimo spremenljivost in nedoločenost zmanjšati je treba najprej razumeti vzroke za njen pojav. Po Shewhartu [15] delimo spremenljivost in variabilnost v delovanju procesa v dve skupini: notranjo in zunanjo.

Notranji vzroki spremenljivosti prihajajo znotraj mej organizacije in jih lahko z ukrepi obvladujemo. Delovni proces, ki se za svoje delovanje poslužuje Kanban sistema, je voden s pravili in politikami, ki jih je vodstvo sprejelo ob implementaciji Kanbana. S spremembami teh pravil neposredno vplivamo na delovanje organizacije. Spremembe pa so lahko motivirane s strani zaposlenih na vseh nivojih organizacije, ne le vodstva. Notranje vzroke za nastanek spremenljivosti nadziramo in odpravljamo s spreminjanjem pravil in politik. Med notranje vzroke spremenljivosti spadajo:

- Velikost (zahtevnost) delovne naloge. Če se ne uporablja standardiziran način določanja vsebine delovnih nalog, lahko njihova velikost oz. obseg zelo niha, kar povzroča nepredvidljivost. Cilj je določati vsebino nalog na tak način, da njihov obseg ne pretirano odstopa od povprečja.
- Razlikovanje nalog glede na vrsto. Če nalog ne razlikujemo glede na vrsto, lahko to privede do večje nedoločenosti velikosti, potrebnega truda za izvedbo, tveganj, in drugih dejavnikov. Z vrsto delovnih nalog mislimo ločevanje na nove funkcionalnosti, odpravo napak in vzdrževalna opravila. Zahteve za naloge posameznih vrst različno pogosto nastopajo in imajo različne prioritete. Z ločevanjem nalog na različne vrste lahko povečamo predvidljivost v sistemu. Drugi način za povečanje predvidljivosti je določitev WIP mej za vsako posamezno vrsto delovnih nalog.
- Razlikovanje nalog glede na razred storitve. Če nalog ne razlikujemo glede na razred storitve, bo posledično spremenljivost sistema večja. Vzemimo za primer organizacijo, ki prejema veliko nujnih zahtev (angl. expedite). Če nujnih nalog ne obravnavamo ločeno, se bo to odražalo na delovnem procesu v povečanju nedoločenosti in daljšanju povprečnega potrebnega časa. To lahko preprečimo z ustrezno velikostjo vhodnega seznama zahtev in ustrezno količino zahtev posameznega razreda storitve, ter z določitvijo WIP omejitev za vsak razred.
- Neenakomeren tok. Ker so v razvoju programske opreme delovne naloge navadno različne ena od druge, prihaja do neenakomernosti v delovnem toku. Neenakomernemu toku se lahko izognemo tako, da vpeljemo izravnalne vrste, in ustrezno povečamo WIP omejitve. Povečanje omejitev bo najverjetneje imelo za posledico podaljšanje povprečnega potrebnega časa, vendar bo hkrati zmanjšalo spremenljivost in umirilo delovni tok. Navadno je večja predvidljivost in bolj konstanten ritem delovnega toka bolj zaželen, kot pa večja prepustnost.
- Predelava funkcionalnosti. Pod predelava funkcionalnosti mislimo dodatno delo zaradi odpravljanja napak in je navadno nenačrtovano. Najboljši način za zmanjšanje spremenljivosti, ki je posledica napak, je ta, da nenehno stremimo k izboljševanju kakovosti. Orodja, ki se za to lahko uporabijo so npr. programiranje v paru (angl. pair programming), "peer review", testi enot (angl. unit tests), uporaba ogrodij za samodejno izvajanje testov (angl. automated testing frameworks), nenehna integracija, majhne izdaje, jasno določene arhitekture sistema, itd.

Zunanji vzroki spremenljivosti izvirajo iz okolja, v katerem organizacija deluje. Na te organizacija nima neposrednega vpliva, lahko se jim le prilagodi. Upravljanje z zunanjimi

vzroki spremenljivosti pokriva področje analize tveganja (angl. risk analysis) in upravljanja s tveganji (angl. risk management). Zunanji vzroki spremenljivosti so naslednji:

- Slabo določene vsebine uporabniških zahtev – tukaj mislimo na dvoumno določene zahteve, slabo definirane poslovne načrte, pomanjkanje strateškega planiranja in vizije. Zaradi pomanjkanja informacij je odločanje oteženo oz. onemogočeno, naloge ni možno dokončati. Da bi se izognili takšnim zaustavitvam delovnega toka, je treba uvesti učinkovit proces za obravnavo in razreševanje težav te vrste.
- Nujne zahteve. To so zahteve, ki se pojavijo nenačrtovano, ob pojavu pa imajo prioriteto izvedbe višjo od ostalih delovnih nalog v teku. Izven organizacije nastanejo, ko kupec odda naročilo ob spremembi zahtev, ki ni bila predvidena. Izvor znotraj organizacije pa so napake v delovnem procesu, zaradi katerih se pozneje v razvoju odkrije pomembne neizdelane funkcionalnosti (npr. težave pri sporazumevanju). Nujne zahteve negativno vplivajo na delovni proces, saj z njihovim nastankom in prioriteto izvedbe zmotijo tok ostalih delovnih nalog in zmanjšajo predvidljivost sistema. Zato so nezaželene in se jim skušamo ogniti, tako da skušamo doseči čim krajši potreben čas (angl. lead time), s povečevanjem transparentnosti in razvojem zrelosti organizacije. Enden od načinov, kako se soočimo z nepredvidljivostjo sistema, ki jo prinašajo nujne zahteve, je ta, da določimo največje število nujnih zahtev, ki se bodo lahko v sistemu hkrati obdelovale.
- Neenakomeren tok. Pojavlja se tako kot notranji, kot tudi kot zunanji vir spremenljivosti. Primer zunanjih virov spremenljivosti so slabo določene zahteve, kadrovska zasedenost, ipd. Navadno pride ob njihovem pojavu do zaustavitve delovnega toka. Ukrepamo lahko na dva načina: povečamo WIP omejitve (gledano na nivoju celotnega delovnega toka) z vpeljavo izravnalnih vrst ali s povečanjem WIP omejitve na aktivnostih. Rezultat tega ukrepa bo povečana stopnja predvidljivosti delovnega toka za ceno daljšega potrebnega časa in potencialno manjše motivacije za izpopolnjevanje delovnega toka. Drugi način je dosledno odkrivanje in razreševanje vzrokov za težave, ki nastajajo. Pri tem bo, vsaj v začetnem obdobju, prihajalo do zastojev v delovnem toku, vendar bodo zaradi izboljšav v delovnem procesu, ki odpravljajo njihov izvor, sčasoma težave vedno bolj redke.
- Delovanje okolij namestitve (angl. deployment environment). Eden od vzrokov za motnje v delovnem toku, prepustnosti in povečanje nepredvidljivosti sistema so tudi motnje v delovanju okolij namestitve. V kanban sistemu so te motnje hitro vidne, saj bo ob njihovem nastanku zaradi upoštevanja omejitev WIP prišlo do zaustavitve v delovnem toku. Člani ekipe, ki delajo na delovnih nalogah pred točko zaustavitve,

bodo v takem primeru ostali brez novih delovnih nalog. Člani ekipe, ki so ostali brez dela, v takem primeru navadno pomagajo odpravljati težavo, čeprav ta ne sega na področje, za katerega so odgovorni.

- Vplivi trga. Na spremenljivost vplivajo razmere trga, na katerem organizacija posluje. Kot primer lahko vzamemo svetovno gospodarsko krizo iz leta 2008, ki je temeljito spremenila razmere na tržišču. Podjetja so se morala novim razmeram prilagoditi. V organizaciji lahko do neke mere takšne razmere predvidimo z uporabo strategij s področja upravljanja s tveganji. Kanban sistemi pri svojem delovanju upoštevajo vrsto elementov, ki pripomorejo k upravljanju s tveganji.
- Težave z izpeljavo koordinacijskih aktivnosti. To so težave s koordiniranjem med organizacijami, interesnimi skupinami in resursi. Pogosto se predvidijo in rešujejo s planiranjem rednih sestankov.

2.3 Sistematični pregled literature

2.3.1 Uvod

Sistematični pregled literature je postopek prepoznavanja, vrednotenja in razlage vseh raziskav, ki so na voljo o določenem raziskovanem vprašanju, področju ali pojavu. Študije, ki so predmet preučevanja v sklopu sistematičnega pregleda, imenujemo primarne študije (angl. primary studies).

Sistematični pregled se lahko izvaja iz več razlogov. Med najpogostejšimi so povzemanje obstoječih odkritij o nekem področju, tehnologiji, iskanje pomanjkljivosti v obstoječih raziskavah in spodbujanje njihovega odpravljanja, ter nudenje podlage za ustrezno načrtovanje novih raziskav.

Prednosti, ki jih prinaša sistematični pregled, so zmanjšanje možnosti za pristranskost obravnave gradiva zaradi uporabe jasno določene metodologije, pridobivanje informacij o obnašanju obravnavanega pojava v različnih okoljih in primerih iz prakse, in povzemanje podatkov po metaanalitičnih metodah (angl. meta-analytic techniques) pri izvajanju kvantitativnih študij. Glavna pomanjkljivost sistematičnega pregleda je ta, da je vanj treba vložiti veliko truda in časa.

Od običajnega strokovnega pregleda se sistematični pregled literature loči v več stvareh. Začne se z določitvijo protokola pregleda v katerem se navede raziskovalna vprašanja in metode dela, ki bodo uporabljene v pregledu. Sistematični pregled literature temelji na določeni strategiji iskanja, katere cilj je odkriti čim več relevantne literature. Strategija iskanja je dokumentirana in tako bralcem omogoča oceniti natančnost, popolnost in ponovljivost procesa. Kriteriji vključitve in izključitve primarnih študij v obravnavo so jasno določeni. Poleg kriterijem vključitve morajo primarne študije za vključitev v obravnavo ustrezati tudi zastavljeni ravni kakovosti, ki se ocenjuje z orodjem za ocenjevanje kakovosti.

Postopek sistematičnega pregleda je razdeljen v tri faze: načrtovanje, izvedba in poročanje o pregledu literature. Faze in aktivnosti znotraj njih so podrobneje opisane v nadaljevanju. Omeniti je treba, da si aktivnosti ne nujno sledijo zaporedoma, kot so v tem delu navedene, ampak se lahko izvajajo v iteracijah - pogosto se nekatere poznejše aktivnosti začnejo že ob definiranju protokola pregleda v fazi načrtovanja, ter se kasneje nadaljujejo v ostalih dveh fazah pregleda.

V nadaljevanju so opisane smernice za izvedbo sistematičnega pregleda literature, kot jih podajata Kitchenham in Charters v [7]. Smernice za izvedbo pregleda na področju razvoja

programske opreme so v [7] nastale na podlagi smernic iz področja zdravstva. V delu se tematiki večkrat prepletata in primerjata. V pričujočem delu se bomo omejili le na področje razvoja programske opreme.

2.3.2 Načrtovanje

Načrtovanje je začetna faza postopka sistematičnega pregleda literature. V tej fazi se najprej preverja smiselnost izvedbe pregleda. Če je potrebno, se pripravi dokument o naročilu pregleda. Najpomembnejša aktivnost te faze je določitev raziskovalnih vprašanj, ki se bodo obravnavala, in izdelava protokola pregleda (angl. review protocol), v katerem se določi osnovne postopke in metode dela, katerih se bomo posluževali v pregledu. Ustreznost protokola pregleda naj bo preverjena neodvisno od izvajalcev pregleda. To je še posebej potrebno v primeru izvajanja pregleda po naročilu, ki ga odda v izvajanje organizacija, ki v danem trenutku nima razpoložljivih resursov (čas, kadri,...).

Izvajanje pregleda po naročilu je izven obsega pričujočega dela in ga ne bomo obravnavali. Podrobneje je predstavljeno v [7].

2.3.2.1 Smiselnost izvajanja sistematičnega pregleda literature

Sistematični pregled literature se izvede zaradi potreb raziskovalcev po celostnem pregledu trenutnega stanja raziskanosti nekega področja, ki ga želijo izvesti temeljito in nepristransko. Raziskovalec mora najprej preveriti ali je izvajanje sistematičnega pregleda zares potrebno – ali morda že obstaja sistematični pregled, ki pokriva področje raziskave. Če se ugotovi, da taki pregledi že obstajajo, se njihovo ustreznost preveri z naslednjimi vprašanji:

- Ali se v pregledu obravnava pomembno vprašanje s področja razvoja programske opreme?
- Ali je bilo iskanje literature izvedeno temeljito in na ustreznih podatkovnih bazah, virih literature?
- Ali je bila kakovost uporabljene metodologije ovrednotena, ter ali je bilo vrednotenje ustrezno?
- V kolikšni meri so rezultati pregleda odvisni od uporabljene metode pri njeni izvedbi?
- Ali so bili numerični rezultati smiselno interpretirani upoštevajoč širši vidik problematike?

Obstajajo tudi druga orodja preverjanja ustreznosti, kot na primer vprašalnik po CRD (angl. Centre for Reviews and Dissemination, Univerza v Yorku) in DARE (angl. Database of Abstracts of Reviews of Effects). Obe orodji sta izven konteksta pričujočega dela in sta opisani v [7].

2.3.2.2 Raziskovalna vprašanja

Določitev raziskovalnih vprašanj (angl. research questions) je najpomembnejši del vsakega sistematičnega pregleda literature. Raziskovalna vprašanja so gonilo celotnega postopka sistematičnega pregleda. Na podlagi tematike in obsega, ki sta z njimi določena, se v procesu iskanja poskuša identificirati ustrezno literaturo (primarne študije), v procesu pridobivanja podatkov (angl. data extraction process) se skuša priti do podatkov, s katerimi bomo lahko na vprašanja odgovorili, proces analize podatkov pa mora pridobljene podatke strniti tako, da predstavljajo odgovor na raziskovalna vprašanja.

Tipi raziskovalnih vprašanj

Predlaga se naslednjih pet tipov raziskovalnih vprašanj:

1. Vprašanja, ki presojujejo učinek tehnologije razvoja programske opreme,
2. Vprašanja, ki ugotavljajo pogostost nastopa nekega pojava znotraj razvojnega procesa (npr. vpeljava nove tehnologije, frekvenca uspešno in neuspešno zaključenih projektov, ipd.),
3. Vprašanja, ki identificirajo stroške in dejavnike tveganja povezane z neko tehnologijo,
4. Vprašanja, ki identificirajo vpliv tehnologij na zanesljivost, zmogljivost in stroškovne modele.
5. Vprašanja, ki se tičejo analize stroškov in koristi uporabe določenih tehnologij razvoja programske opreme ali aplikacij.

Ključnega pomena je, da se v pregledu postavi prava vprašanja. Ta naj bodo pomembna, pomenljiva in zanimiva tako za raziskovalce, kot tudi za ljudi, ki se s tematiko ukvarjajo v praksi. Z njimi naj se poskuša vpeljati spremembe v trenutno uveljavljene prakse ali pa okrepiti zaupanje v pravilnost uveljavljenih praks. Razkrivajo naj razhajanja med splošno uveljavljenimi prepričanji in predvidevanji na eni ter dejanskim (realnim) stanjem na drugi strani.

Struktura raziskovalnih vprašanj

Raziskovalna vprašanja naj bodo sestavljena po metodi PICOC, ki je akronim za pojme populacija, poseg, primerjava, izid, in okvir (angl. Population, Intervention, Comparison, Outcome, Context).

Populacija (angl. Population)

Pod pojmom populacija mislimo na ljudi, ki se jih raziskovana tematika zadeva:

- Funkcija, oz. vloga, npr. testerji, razvijalci, vodje,
- Izkušnost: začetnik ali izkušen inženir,
- Področje uporabe, npr. IT sistemi, računalniška varnost,
- Gospodarsko področje, npr. podjetje, ki se ukvarja s telekomunikacijami, majno IT podjetje, ipd.

Ni nujno, da se z enim raziskovalnim vprašanjem omejimo na natanko en segment populacije. Eno samo raziskovalno vprašanje lahko pokriva tudi več segmentov, npr. zanimajo nas testerji začetniki ALI izkušeni arhitekti programske opreme.

Poseg (angl. Intervention)

Pod poseg mislimo na metodologijo, orodje, tehnologijo ali postopek, ki rešuje neko konkretno problematiko. To je npr. tehnologija za izvajanje določene naloge, kot so priprava uporabniških zahtev, testiranje sistema, ipd.

Primerjava (angl. Comparison)

S primerjavo mislimo na metodologijo, orodje, tehnologijo ali postopek, s katerim se poseg primerja.

Izid (angl. Outcome)

Izidi naj se nanašajo na za tematiko vprašanja pomembne dejavnike, kot so povečana zanesljivost, znižani stroški izdelave, ipd. Če se vprašanje nanaša na več dejavnikov, naj bodo navedeni vsi.

Okvir (angl. Context)

Pod tem pojmom razumemo okvir, oz. kontekst populacije - udeležencev študije (akademiki, študenti, svetovalci), primerjave (akademska sfera, gospodarstvo), in posega (majhne

razsežnosti, velike razsežnosti). Na primer, vprašanje se nanaša na akademsko področje, udeleženci so študenti, ter naloge so majhnih razsežnosti.

2.3.2.3 Izdelava protokola pregleda

Sistematični pregled literature mora potekati nepristransko, raziskovalec s svojimi pričakovanji ne sme vplivati na vključitev primarnih študij v raziskavo. To se zagotovi s protokolom pregleda, s katerim se določi pravila izvedbe pregleda, in mora biti izdelan že pred začetkom faze izvedbe pregleda.

V protokolu pregleda so določene metode, ki se bodo uporabile za izvedbo sistematičnega pregleda. To naj bi zavzemalo vse elemente pregleda in še nekaj dodatnih informacij:

- Ozadje, razlogi za pregled,
- Raziskovalna vprašanja, na katera se bo v pregledu odgovarjalo,
- Strategija iskanja primarnih študij, vključno z iskalnimi nizi in viri, ki bodo preiskani (digitalne knjižnice, revije, zborniki, itd.).
- Kriteriji za vključitev študij. Kriteriji določajo katere primarne študije bodo upoštevane v pregledu in katere ne.
- Postopek za vključitev študij. Postopek določa kako se bodo kriteriji za vključitev aplicirali na seznamu najdenega gradiva, ter kako se bodo razreševala nesoglasja raziskovalcev pri vključevanju primarnih študij v pregled.
- Seznam kriterijev in postopek ocenjevanja kakovosti študije. Raziskovalci morajo sestaviti seznam kriterijev, s katerim bodo ocenjevali ustreznost študije za vključitev v pregled.
- Strategija pridobivanja podatkov (angl. data extraction strategy) določa kako se bo pridobivalo podatke iz primarnih študij. Če je treba podatke preoblikovati, prevesti ali iz njih sestavljati domneve, je treba definirati protokol validacije podatkov.
- Sinteza podatkov (angl. synthesis of the extracted data) definira strategijo interpretiranja pridobljenih podatkov. Razjasni se, ali bo potrebno izvesti formalno metaanalizo, ter katere tehnike bodo pri tem uporabljene.
- Strategija objave rezultatov (angl. dissemination strategy).
- Časovnica projekta, ki določa časovni načrt pregleda.

2.3.2.4 Presoja ustreznosti protokola pregleda

Protokol pregleda je ključni element sistematičnega pregleda literature. Raziskovalci morajo določiti postopek, s katerim se oceni ustreznost protokola. V ta namen se lahko prilagodi vprašalnik, ki smo ga omenili zgoraj in je namenjen ugotavljanju smiselnosti izvajanja sistematičnega pregleda literature. Dodatno naj se preveri še: da so iskalni nizi ustrezno izpeljani iz raziskovalnih vprašanj, da bodo najdeni podatki ustrezali raziskovalnim vprašanjem, in da je postopek analize podatkov primeren za odgovarjanje na raziskovalna vprašanja.

2.3.3 Izvedba pregleda

Ko je protokol pregleda določen, se lahko začne izvedba pregleda. Kot smo že omenili, se nekatere aktivnosti faze izvedbe pregleda pogosto delno izvajajo že v fazi načrtovanja pri določanju protokola pregleda.

2.3.3.1 Opredelitev raziskave

Cilj sistematičnega pregleda je poiskati kar največ primarnih študij, ki pokrivajo tematiko, opredeljeno z raziskovalnimi vprašanji. Iskanje naj bo nepristransko in temeljito. S tem se sistematični pregled tudi loči od tradicionalnega načina izvajanja pregleda literature.

Na začetku je treba opredeliti strategijo iskanja. V pomoč pri tem lahko izvedemo predhodna iskanja, poskusna iskanja, primerjamo iskalne nize s tistimi v že obstoječih, sorodnih sistematičnih pregledih, ter se posvetujemo s strokovnjaki s področja.

Namen predhodnega iskanja (angl. preliminary search) je odkriti, ali že obstaja sistematični pregled z raziskovano vsebino ter po drugi strani oceniti obseg za raziskavo primerne literature.

Poskusna iskanja izvajamo tako, da za iskalni niz uporabljamo različne kombinacije izrazov, ki izhajajo iz raziskovalnih vprašanj.

Običajen pristop je, da se raziskovalna vprašanja razbije na posamezne sklope PICOC. Za vsakega izmed teh sklopov se določi seznam iskalnih izrazov, ki ga sestavljajo tudi sopomenke, blizuzvočnice in kratice. Z logičnima operatorjema AND in OR lahko sestavimo tudi bolj kompleksne izraze.

Predhodna in poskusna iskanja lahko izvajamo na digitalnih knjižnicah. Poleg teh pa je treba pri sistematičnem pregledu upoštevati tudi druge vire literature, kot so: sezname referenčne

literature relevantnih primarnih študij, revije, zbornike in sivo literaturo, raziskovalne registre, ter ne nazadnje Internet.

Pri iskanju literature je treba biti pozorni na pristranskost publikacije (angl. publication bias). Upoštevati je treba, da obstaja večja verjetnost objave dela, ki razkriva pozitivne (pričakovane) rezultate, kot pa dela z negativnimi rezultati. Posebej pozorni je treba biti v primerih, ko so študije izvedene po naročilu. Naročnik študije bo malo verjetno objavil študijo z rezultati, ki ne ustrezajo njegovemu poslovanju. Pristranskosti publikacij se lahko ognemo s prijemi, ki smo jih opisali v sklopu določitve strategije iskanja, poleg tega pa še s pregledom sive literature, zbornikov konferenc, ter tako, da poizvemo o neobjavljenih študijah neposredno pri raziskovalcih in strokovnjakih na obravnavanem področju.

Ko uspemo zbrati seznam študij, ki bodo potencialno uporabne, je treba gradivo pridobiti. Pri tem je dobro uporabiti sistem za sledljivost, tako da ne bomo izpustili kakšne pomembne študije.

Postopek izvajanja sistematičnega pregleda literature naj bo transparenten in karseda ponovljiv. Pregled mora biti dokumentiran v taki meri, da omogoči bralcu ocenitev temeljitosti njegove izvedbe. Vse spremembe v iskanju naj bodo zabeležene in utemeljene. Rezultati iskanja (brez upoštevanja filtrov iskanja), naj bodo shranjeni in na voljo za morebitno kasnejšo ponovno analizo.

Pri dokumentiranju iskanja po različnih virih naj bo zabeleženo vsaj naslednje:

- digitalne knjižnice: zabeležimo ime podatkovne baze, strategijo iskanja po podatkovni bazi, datum iskanja, omejitev po letih (od, do).
- Ročni pregled revij: ime revije, omejitev po letih, izdaje, ki niso bile pregledane.
- Zborniki konferenc: naslov zbornika, naziv konference (če se razlikuje od naslova zbornika), prevod naslova (po potrebi), ime revije (v primeru objave v reviji).
- poskus odkrivanja študij, ki niso bile objavljene: Raziskovalne skupine in raziskovalci, ki smo jih kontaktirali (imena in kontaktni podatki), raziskovalna spletna mesta, ki smo jih pregledali (datum in URL),
- drugi viri literature: datum iskanja, URL, kakršnekoli posebne okoliščine iskanja.

Raziskovalci naj v dokumentaciji iskanja navedejo razloge za izbiro digitalnih knjižnic, ki se bodo pregledale, izbiro revij in zbornikov, ki se bodo pregledali, ter izbiro načina izvajanja iskanja – ročnega, ali elektronskega, ali kombinacijo obeh.

2.3.3.2 Izbira relevantnih študij

Z izvedbo iskanja se pridobi študije, za katere je treba oceniti relevantnost glede na raziskovano področje. V ta namen je treba definirati kriterije za vključitev in izključitev študij za obravnavo v pregledu, t.j. kriteriji s katerimi se poskuša ugotoviti povezavo vsebine študije z raziskovalnim vprašanjem. Kriterije naj se definira že ob izdelavi protokola pregleda, po potrebi se dopolnijo med izvedbo sistematičnega pregleda. Na ta način se izognemo pristranskosti kriterijev.

Izbira primarnih študij poteka v več korakih. V prvem koraku naj se ustreznost kriterijem vključitve in izključitve preverja v naslovih in povzetkih primarnih študij. Če naslov in povzetek nudita premalo informacij za sprejem odločitve o vključitvi ali izključitvi študije, je treba predelati celotno besedilo študije.

V naslednjem koraku naj se preverjajo kriteriji za vključitev ali izključitev študije, ki temeljijo na bolj praktičnih zadevah, kot so jezik, revija, avtorji, umestitev študije, udeleženci študije, načrt raziskave (angl. study design), metoda vzorčenja in datum objave.

Tretji korak izbire študij se izvaja po potrebi. V tem koraku raziskovalci preverjajo ustreznost študij glede na kriterije kakovosti.

Priporoča se, da se hrani tudi seznam izključene literature. Ker pa iskanja literature navadno odkrijejo tudi študije, ki so popolnoma nepovezane s tematiko pregleda, je dobro, da take študije najprej izločimo, šele nato pa začnemo voditi seznam izključene literature.

V primeru, da sistematični pregled izvajata dva ali več raziskovalcev, je treba skladnost odločitve o vključitvi ali izključitvi študije meriti s statistično meritvijo Cohen Kappa. Vsaka meritev naj se dokumentira v končnem poročilu. Vsako nesoglasje je treba obravnavati in ga razrešiti. Študije, za katere se raziskovalci ne morejo odločiti o vključitvi ali izključitvi, naj se preverijo s tehniko analize občutljivosti. Ko sistematični pregled izvaja en sam raziskovalec, lahko da seznam vključene in izključene literature v pregled mentorju, strokovnjaku s področja, ali drugemu raziskovalcu, lahko pa zanesljivost odločitve preveri z metodo test-retest, ter ponovno preverjanje kriterijev za vključitev/izključitev naključno izbrane študije.

2.3.3.3 Ocena kakovosti študije

Poleg upoštevanja kriterijev za vključitev primarnih študij je treba pred uvrstitvijo študije v obravnavo v pregledu tudi oceniti njeno kakovost. Ocena kakovosti študij določa dodatne, bolj podrobne kriterije, na podlagi katerih se sprejme odločitev za vključitev ali izključitev študij. Na podlagi ocene kakovosti lahko ugotovljamo, ali so razlike v rezultatih študij odraz razlik v njihovi kakovosti, ter pomembnost posameznih študij, v primeru ko se rezultati več študij povzemajo (angl. *synthesise studies*). Kakovost študije lahko uporabljamo kot vodilo za interpretacijo ugotovitev, določanja zanesljivosti sklepanja v študijah, ter nadaljnje raziskovanje.

Znanstveniki si sicer niso enotni v definiciji pojma kakovosti študije, nekateri pa so mnenja, da nam kakovost študije pomeni mero, v kateri študija minimizira pristranskost in maksimira notranjo in zunanjo veljavnost. Pod pristranskost je tukaj mišljen odmik rezultatov študije od "pravih" rezultatov, notranja veljavnost nam pove v kakšnem obsegu zasnova in izpeljava študije preprečujeta sistematične napake, zunanja veljavnost pa nam pove v kakšnem obsegu so učinki, ki so obravnavani v študiji, primerni za uporabo izven študije.

Na kakovost študije vpliva tudi vrsta znanstvenega poskusa, ki se v študiji uporablja. Tako znanstveniki predlagajo upoštevanje hierarhije pridobljenih dokazov, pri čemer so na vrhu hierarhije dokazi iz sistematičnih pregledov in randomiziranih nadzorovanih raziskav, na dnu pa rezultati kvazi-raziskav in izvedenska mnenja. Z uporabo te hierarhije dokazov se lahko raziskovalec pri izvajanju sistematičnega pregleda omeji le na določene vrste študij. Izbira vrst študij naj ustreza postavljenim raziskovalnim vprašanjem.

Razvoj orodij za oceno kakovosti študij

Orodje za oceno kakovosti študij je seznam dejavnikov, ki jih je treba ovrednotiti za vsako študijo. Če tem dejavnikom priredimo številčne vrednosti, lahko kakovost študije numerično ovrednotimo. Če želimo pri ocenjevanju kakovosti študij uporabiti mersko lestvico, lahko odgovore tipa Da/Ne nadomestimo s številčnimi odgovori, npr. od 1 do 5. Na seznam so uvrščeni tisti dejavniki, ki bi lahko privedli do pristranskosti rezultatov študije. Upoštevati je treba tudi težave, ki jih povzroča pristranskost in veljavnost na različnih nivojih empirične študije: načrt, izvedba, analiza in zaključki. V [7] so avtorji podali primer orodij za ocenjevanje kvantitativnih in kvalitativnih študij. Ob izvajanju sistematičnega pregleda je treba seznam vprašanj prilagoditi potrebam pregleda in iz njega izbrati le vprašanja, ki so za pregled smiselna.

Orodje za oceno kakovosti študije navadno izdelamo že v fazi načrtovanja ob oblikovanju protokola pregleda, dokončano pa naj bo do začetka izvajanja ocenjevanja kakovosti študij. V okviru oblikovanja protokola pregleda naj se določi tudi način uporabe orodja za preverjanje kakovosti študij. Ko je enkrat izdelano, je treba preveriti in presoditi njegovo uporabnost in zanesljivost.

Oceno kakovosti študije, ki jo dobimo z orodjem, lahko uporabimo:

1. V namen olajšanja sprejema odločitve za vključitev ali izključitev primarne študije v raziskavo pregleda. V tem primeru nam orodje za ocenjevanje kakovosti študije nudi razširitev kriterijev za vključevanje / izključevanje študij, ti so navadno bolj podrobni. Podatke o kakovosti posamezne študije, ki jih pridobimo z uporabo orodja, je treba zbirati ločeno od postopka pridobivanja podatkov. Navadno se za to uporabljajo posebni obrazci.
2. V pomoč pri analizi in sintezi podatkov. V tem primeru z orodjem za ocenjevanje kakovosti pridobimo neke vrste povzetek primarne študije, ki nam pomaga pri ugotavljanju, ali so razlike v rezultatih raziskav odvisni od njihove kakovosti. Ocenjevanje kakovosti lahko v tem primeru poteka hkrati s postopkom pridobivanja podatkov, obrazci orodja za oceno kakovosti so lahko združeni z obrazci za pridobivanje podatkov.

Seveda lahko orodje za ocenjevanje kakovosti v okviru enega sistematičnega pregleda uporabljamo na oba načina.

V primeru, da v okviru sistematičnega pregleda obravnavamo različne tipe študij (revija, zbornik, siva literatura, ...) je treba za vsak tip študije izdelati svoje orodje za ocenjevanje kakovosti. Lahko pa se uporabi tudi eno samo orodje za ocenjevanje kakovosti za vse tipe študij, ki jih sistematični pregled obravnava, če se to oceni kot ustrezno. Če se v pregledu obravnava kvalitativne in kvantitativne študije, je nujno treba uporabiti različna orodja za oceno kakovosti.

Pri uporabi orodij za ocenjevanje kakovosti se je treba zavedati tudi njihovih omejitev. Včasih v primarni študiji ne najdemo odgovora na vprašanje orodja za ocenjevanje kakovosti. Ko nek podatek ni zapisan v poročilu primarne študije radi predpostavimo, da raziskovalci tematike niso obravnavali, kar ni nujno res. Orodje za ocenjevanje kakovosti naj ocenjuje kakovost metodologije raziskave v študiji, ne pa samega poročila študije. V takem primeru bi bilo dobro, da raziskovalec poskuša pridobiti čim več podatkov o študiji, če je možno, tudi od avtorjev samih.

2.3.3.4 Pridobivanje podatkov

Glavni cilj te aktivnosti je oblikovanje obrazcev, s katerimi bodo lahko raziskovalci z dovoljšno mero natančnosti pridobivali podatke iz primarnih študij. Obrazci za pridobivanje podatkov naj bodo narejeni in pilotno preverjeni v fazi načrtovanja ob oblikovanju protokola pregleda. S tem se izognemo možnosti za pristransko obravnavo.

Obrazci za pridobivanje podatkov morajo biti zasnovani tako, da bo s pridobljenimi podatki možno odgovoriti na raziskovalna vprašanja ter na kriterije orodja za ocenjevanje kakovosti študije. V večini primerov bo obrazec za pridobivanje podatkov vseboval množico numeričnih vrednosti, ki jih je potrebno pridobiti iz študij. Numerični podatki omogočajo povzemanje in sintezo podatkov in so osnova za tehnike s področja statistike, ki se uporabljajo za strnjevanje rezultatov primarnih študij.

Ko je oblikovanje obrazcev za pridobivanje podatkov končano, jih je treba testno uporabiti nad vzorcem primarnih študij. Tako bomo lahko odkrili in odpravili napake vsebine obrazcev, ter preverili uporabnost obrazcev (razumljivost navodil za uporabo, urejenost vprašanj, ipd.). Če bo obrazce uporabljalo več raziskovalcev, je dobro, da pri testni uporabi obrazcev sodelujejo vsi. Opravilo si lahko poenostavimo z elektronskimi obrazci, ki olajšujejo tako zbiranje podatkov, kot tudi nadaljnje analize zbranih podatkov.

Obrazci za pridobivanje podatkov naj poleg vprašanj, ki so potrebna za pridobivanje podatkov za tvorjenje odgovora na raziskovalna vprašanja, vsebujejo še vsaj podatke o imenu in priimku izvajalca pridobivanja podatkov, datum pridobivanja podatkov, naslov, avtorje, ime revije in podrobnosti o izdaji, ter določen naj imajo prostor za dodatne zabeleške.

Ko je to mogoče, naj pridobivanje podatkov iz primarnih študij izvajata dva ali več raziskovalcev. Podatki, ki jih raziskovalci pridobijo, naj bodo med sabo primerjani, nestrinjanja raziskovalcev pa odpravljena. Če so zaradi časovne, kadrovske ali kakšne druge stiske raziskovalci primorani obravnavati vsak svoje primarne študije, je treba zagotoviti, da to delajo na konsistenten način. Nekatere (naključno izbrane) študije naj bodo zato obdelane s strani vseh raziskovalcev.

Če raziskavo opravlja en sam raziskovalec, naj bodo rezultati pridobivanja podatkov preverjeni s strani mentorja ali drugega strokovnjaka. Ta naj pridobivanje podatkov izvede na naključno izbrani študiji, rezultati pa naj se primerjajo. Druga možnost je, da raziskovalec sam ponovno izvede pridobivanje podatkov na naključno izbrani študiji, ki je že bila obdelana.

Pri izvajanju sistematičnega pregleda moramo biti posebej pozorni, da med izbrane primarne študije ne vključimo več različnih poročil iste raziskave, saj lahko to bistveno pokvari rezultate pregleda.

Opozoriti je treba še na tri težave, ki jih povzročajo: neobjavljeni, manjkajoči in podatki, ki zahtevajo dodatno obdelavo. Če je raziskava še v teku, je lahko poročilo raziskave, oz. študija še v izdelavi in tako ni na voljo v celoti. V takem primeru lahko ustrezne podatke pridobimo od avtorja študije. Poročila raziskav, ki jih v pregledu obravnavamo kot primarne študije, ne vsebujejo vedno vseh za nas pomembnih podatkov. Tudi v tem primeru poskušamo potrebne podatke pridobiti od avtorjev raziskave. Včasih lahko z dodatno obdelavo podatkov, ki so navedeni v poročilu raziskave, pridemo do podatkov (zaključkov), ki v poročilu sicer niso navedeni. Takrat je treba podatke najprej prikazati na enak način, kot so bili podani v primarni študiji, šele nato pa jih obdelati. Obdelani podatki morajo biti dodatno preverjeni z analizo občutljivosti.

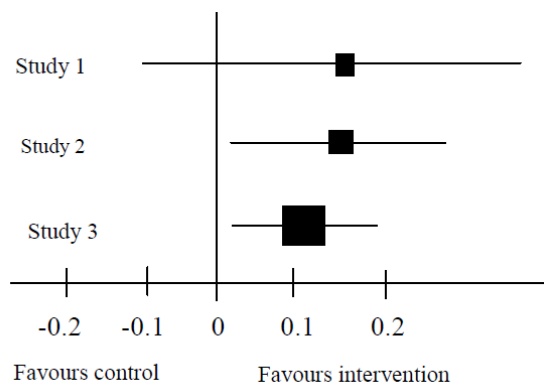
2.3.3.5 Sinteza podatkov

Sinteza podatkov vključuje primerjavo in povzemanje rezultatov primarnih študij, ki so bile vključene v sistematični pregled. Lahko je opisna (ne-kvantitativna), kvalitativna ali kvantitativna. Opisna sinteza podatkov je lahko dopolnjena s kvantitativnim povzetkom, ki je opravljen z uporabo statističnih tehnik (metaanaliza). Koraki, ki so potrebni za izvedbo sinteze podatkov naj bodo navedeni v protokolu pregleda.

Pri opisni sintezi podatkov naj bodo iz primarnih študij pridobljene informacije predstavljene v tabelah. Tabelarni prikaz naj bo zasnovan tako, da ustreza raziskovalnemu vprašanju in se poudarja podobnosti in razlike med posameznimi primarnimi študijami. Odkriti želimo, ali so raziskovalci v posameznih primarnih študijah prišli do podobnih rezultatov (homogenost rezultatov) ali se njihove ugotovitve razlikujejo (heterogenost rezultatov). Tabelarni prikaz podatkov v sintezi lahko zasnujemo tudi tako, da je možno razbrati vzroke heterogenosti rezultatov (tip študije, kakovost študije, velikost vzorca, ipd.).

Podatki kvantitativne sinteze naj bodo prav tako predstavljeni v tabelarni obliki. Pri tem je treba podati vsaj: velikost vzorca vsakega posega, oceno obsega učinka in standardno napako za vsak učinek, razliko med sredinskimi vrednostmi (angl. mean values) za vsak poseg, ter interval zaupanja (angl. confidence interval) za vsako razliko, ter merske enote, ki so bile uporabljene za merjenje vsakega posameznega učinka. Če želimo izvesti kvantitativno sintezo nad podatki različnih študij, je treba rezultate študij prikazati na tak način, da bodo med sabo primerljivi. Pri tem se predlaga, da za različne tipe rezultatov uporabimo različne mere učinkov.

Najpogosteje se za vizualizacijo kvantitativnih rezultatov uporabljajo drevesni diagrami (angl. forest plot). Ti predstavljajo povprečne odmike in variance za vsako študijo. Črta predstavlja standardno napako razlike, kvadrat pa povprečno vrednost razlike. Velikost kvadrata ustreza številu subjektov v študiji.



Slika 6: Primer drevesnega diagrama za prikaz kvantitativnih rezultatov [7].

Za binarne rezultate (npr. da / ne) lahko uporabljamo različne mere: kvote (angl. odds), tveganje (angl. risk), relativne kvote (angl. odds ratio, OR), relativno tveganje (angl. relative risk) in absolutno zmanjšanje tveganja (angl. absolute risk reduction). Pod kvote mislimo na razmerje med številom primerkov v vzorcu, ki imajo neko lastnost, in številom primerkov, ki lastnosti nimajo. Relativno tveganje je razmerje med številom primerkov v vzorcu, ki imajo neko lastnost, in skupnim številom primerkov v vzorcu. Relativne kvote so razmerje med kvotami v testni in kvotami v kontrolni skupini. Relativno tveganje je razmerje med tveganjem v testni in tveganjem v kontrolni skupini. Absolutno zmanjšanje tveganja je absolutna razlika med pogostostjo dogodka med primerjalnimi skupinami (angl. comparison groups).

Za merjenje zveznih podatkov se lahko uporabi mere:

- povprečna razlika (angl. mean difference, oz. difference in means) je razlika med povprečji skupin (kontrolnih in testnih).
- tehtana povprečna razlika (angl. weighted mean difference) – ko se za razlike uporablja ista merska lestvica za vse študije, je utež, ki se jo določi vsaki študiji izračunana po metodi inverzne variance.
- standardizirana povprečna razlika (angl. standardised mean difference). Pogosta težava pri povzemanju rezultatov raziskav je, da so rezultati predstavljeni na različne načine, z različnimi merami. V takem primeru lahko povprečno razliko delimo z

ocenjenim standardnim odklonom znotraj posameznih skupin, ter tako pridobimo standardizirano vrednost brez enot. SMD lahko upoštevamo le v primeru, ko razlike v standardnem odklonu odražajo le razlike na merski lestvici in ne tudi dejanskih razlik med testnimi vzorci.

Pri kvantitativni sintezi rezultatov želimo povzeti zaključke in rezultate študij, ki so predstavljeni z besedo, pri čemer je treba upoštevati možnost različne uporabe pojmovanja s strani raziskovalcev. K sintezi kvantitativnih rezultatov lahko pristopamo na tri načine:

- vzajemna pretvorba (angl. reciprocal translation). Ko študije obravnavajo sorodno področje, lahko sintezo podatkov izvedemo tako, da vsak primer "pretvorimo" v vsakega od ostalih,
- Sinteza zavračanj (angl. refutational synthesis). Ko se študije posredno ali neposredno zavračajo med sabo, je treba pretvoriti tako posamezne študije, kot tudi ovračanja, kar nato omogoča podrobnejšo obravnavo zavračanj,
- Sinteza argumentov (angl. line of argument synthesis). Uporablja se, ko izbrane študije pokrivajo le dele tematike, na podlagi katerih bi radi v sintezi prišli do sklepanj v okviru celotne tematike. Sinteza se izvaja v dveh korakih. V prvem se analizira posamezne študije, v drugem pa se poskuša analizirati množico študij, ki naj bi predstavljala zaključeno celoto.

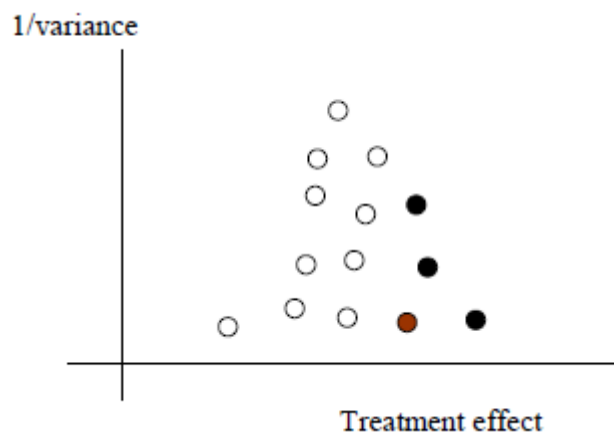
Ko imamo v sistematičnem pregledu opravka s sintezo obeh vrst, kvalitativnih in kvantitativnih študij, je treba najprej izvesti ločeni sintezi. Šele nato se poskuša sintezi združiti tako, da kvalitativni rezultati nudijo dodatno razlago kvantitativnih rezultatov.

Naslednji korak po opravljeni sintezi podatkov je analiza občutljivosti. To je najlažje opraviti v okviru metaanalize, kjer je treba rezultate analize ponoviti na več podmnožicah primarnih študij, da se lahko oceni robustnost rezultatov. Primarne študije lahko združujemo v podmnožice tako, da izberemo:

- primarne študije, z najvišjo oceno kakovosti,
- primarne študije določenih tipov,
- primarne študije, pri katerih je postopek pridobivanja podatkov potekal brez težav (konfliktov),
- glede na raziskovalno metodo, ki je bila uporabljena v primarni študiji.

Za opisno sintezo podatkov, je ocena občutljivosti nedvomno bolj subjektivna, vendar je vseeno dobro preveriti kakšen vpliv ima na rezultate raziskave, če se iz obravnave izvzame študije z nižjo oceno kakovosti, ali npr. študije določenega tipa.

Pomemben korak faze pridobivanja podatkov je preverjanje občutljivosti sistematičnega pregleda na pristranskost objavljanja (angl. publication bias). Za preverjanje se uporablja lijakasti prikaz inverzne variance (ali velikosti vzorca) v odvisnosti od učinka intervencije, t.j. povprečne razlike med testno in kontrolno skupino. Za sistematični pregled, ki ima lijakasti prikaz simetrične oblike lahko rečemo, da ne bo kazal prisotnosti pristranskosti objavljanja. Lijakasti prikaz asimetrične oblike pa nakazuje na prisotnost pristranskosti objavljanja obravnavanih primarnih študij. Slika 7 prikazuje primer lijakastega prikaza. Ker je oblika grafa simetrična, lahko sklepamo, da pristranskost objavljanja ni prisotna. Če pa v pregledu ne bi bile obravnavane primarne študije, ki so na grafu prikazane z obarvanimi krogi, bi bil graf očitno asimetričen in bi potemtakem namigoval na prisotnost pristranskosti objavljanja obravnavanih študij.



Slika 7: Primer lijakastega prikaza [7].

2.3.4 Poročanje

Zadnja faza sistematičnega pregleda literature je priprava poročila pregleda ter njegova objava oz. distribucija ciljni publikli.

Prva aktivnost je določitev strategije širitve oz. razpečave poročila pregleda. Običajno strategijo širitve vsaj okvirno zastavimo že ob samem začetku pregleda. Če gre za akademsko raziskavo, se ta navadno objavi v reviji ali zborniku, če pa so ciljna publika ljudje, ki se v praksi ukvarjajo s tematiko pregleda, je treba razmišljati tudi o drugih oblikah distribucije, kot

so poljudnoznanstvene revije, javni mediji, propagandni letaki, oglasni plakati, spletne strani ali celo neposredni kontakt ciljne publike.

Oblika poročila naj ustreza namenu objave. Najpogosteje se sistematični pregledi izvajajo v okviru doktorske disertacije ali z namenom objave v reviji ali zborniku. V takem primeru sta obseg in oblika določena. V prilogi Priloga I – Predlog oblike poročila sistematičnega pregleda literature je navedena predlagana oblika poročila sistematičnega pregleda, ki je primerna za tehnična poročila in revije. Poglavlja, ki navadno ne pridejo v poštev pri doktorski disertaciji so označena z zvezdico.

Delo pridobi na veljavi, če je pregledano s strani strokovnjakov. V pomoč pri pregledu nam lahko služi tudi orodje za preverjanje smiselnosti izvedbe sistematičnega pregleda iz poglavja 2.3.2.1.

Poglavje 3 Kanban pristop, med agilnostjo in vitkostjo: sistematični pregled

3.1 Uvod

V tem poglavju bomo predstavili članek Kanban pristop, med agilnostjo in vitkostjo: sistematični pregled [6]. Članek sta objavila avtorja Al-Baik in Miller leta 2014. Avtorja sta v članku objavila poročilo sistematičnega pregleda literature na temo implementacije Kanban pristopa v organizacijo. Namen sistematičnega pregleda je bil razjasniti vodila in elemente Kanban pristopa (pozor na veliko začetnico), cilj pa spodbuditi uspešno vpeljavo Kanban pristopa na področje razvoja programske opreme z jasnim definiranjem primarnih elementov Kanban pristopa in predstavitev prednosti in slabosti uporabe Kanban pristopa v IT organizacijah.

Raziskava je temeljila na 37 primarnih študijah. Po preučitvi le-teh sta avtorja ugotovila, da se jih lahko glede na obravnavanje pojma Kanban razdeli v štiri glavne skupine: 1) Kanban kot element vitkega razvoja, 2) Kanban kot agilna metodologija, 3) hibrid Kanban in Scrum metodologije, in 4) Kanban kot metoda.

3.2 Določitev problematike in obsega

Avtorja na podlagi različnega interpretiranja Kanban pristopa med strokovnjaki ugotavljata, da čeprav se Kanban čedalje več uporablja v podjetjih, ki se ukvarjajo z razvojem programske opreme, ustaljenih in priznanih praks za njegovo uvedbo še ni. S študijo sta želela raziskati primere dobrih praks za implementacijo Kanban na področje razvoja programske opreme. v njej podrobneje opisujeta različne elemente Kanbana, s čimer želita prispevati k njihovemu razumevanju. S študijo sta poskušala tudi postaviti temelje za razvoj smernic za uvedbo Kanban pristopa v organizacije, ki se ukvarjajo z razvojem programske opreme.

Obseg študije sta avtorja omejila na implementacijo Kanban pristopa kot procesa razvoja na področju programske opreme. Kanban pristop (angl. Kanban approach) avtorja definirata kot množico pojmov, načel, praks, tehnik in orodij za upravljanje razvojnega procesa s poudarkom na sprotni dostavi vrednosti kupcem, in s spodbujanjem nenehnega učenja in izboljšav.

Avtorja ugotavljata, da so bili v preteklosti že opravljeni sorodni sistematični pregledi literature, ki pa so se po obsegu in ciljih precej razlikovali. V [16] sta avtorja raziskovala empirične študije na temo agilnih metod in vitkega razvoja programske opreme, ki so bile objavljene do leta 2005. Od 36 obravnavanih primarnih študij, je le ena pokrivala področje vitkega razvoja v procesu razvoja programske opreme. V [17] so se avtorji za vir primarnih študij omejili na poročila primerov iz prakse, ki so bila objavljeni v zbornikih konferenc o agilnosti med leti 2000 in 2011, raziskava pa je bila splošna. Obravnavali so več kot 30 pojmov, praks in načel vitkosti na področju agilnega razvoja programske opreme, niso pa obravnavali vprašanja "Kaj je Kanban?".

3.3 Metodologija raziskave

Avtorja sta raziskavo opravila v skladu s smernicami za izvedbo sistematičnega pregleda literature avtorjev Kitchenham in Charters, ki smo jih opisali v poglavju

Sistematični pregled literature. Faze, aktivnosti in izdelki sistematičnega pregleda literature so predstavljeni v tabeli Tabela 1.

Faza	Aktivnosti	Izdelki
Načrtovanje	- Določitev ciljev pregleda	- Dokument protokola pregleda
	- Določitev raziskovalnih vprašanj	
	- Izdelava in preveritev protokola pregleda	- Obrazec za pridobivanje podatkov
Izvedba	- Določitev iskalne strategije in iskalnih nizov	- Dokument iskalne strategije
	- Določitev virov primarnih študij	- Orodje za ocenjevanje kakovosti
	- Ocenitev kakovosti in izbira primarnih študij za obravnavo (vključitev)	
	- Pridobivanje podatkov iz primarnih študij	- (Preverjen) Obrazec za pridobivanje podatkov
	- Sinteza pridobljenih podatkov	- Povzetek rezultatov
Poročanje	- Predstavitev rezultatov	- Poročilo sistematičnega pregleda
	- Izdelava in oblikovanje poročila sistematičnega pregleda	
	- Ocenitev sistematičnega pregleda	

Tabela 1: Faze, aktivnosti in izdelki sistematičnega pregleda literature.

3.3.1 Cilji sistematičnega pregleda literature

Kot smo že v uvodu v poglavje omenili, je bil cilj sistematičnega pregleda literature razjasniti vodila in elemente Kanban pristopa, da bi spodbudili uspešno vpeljavo Kanban pristopa na

področje razvoja programske opreme z jasnim definiranjem osnovnih pojmov Kanban pristopa in predstavitev prednosti in slabosti uporabe Kanban pristopa v IT organizacijah.

Raziskovalna vprašanja so bila zgrajena po principu PICOC. Za populacijo sta avtorja določila področje uporabe Kanban pristopa, t.j. IT področje razvoja programske opreme. Za poseg sta določila vpeljavo Kanban pristopa v proces razvoja programske opreme. Za primerjavo sta izbrala agilne metodologije (v splošnem), in Scrum (konkretno). Za izid sta določila razjasnitev vodil za implementacijo Kanban pristopa v IT organizacijo z namenom zmanjšanja stroškov in potrebnega časa izdelave izdelkov, ter povečanja kakovosti izdelkov in zadovoljstva naročnikov.

Za raziskovalna vprašanja in podvprašanja sistematičnega pregleda sta avtorja določila naslednje:

RQ 1. Kateri elementi Kanban pristopa so v literaturi obravnavani kot za organizacijo koristni?

RQ 1.1. Kateri primarni elementi Kanban pristopa so obravnavani v literaturi?

RQ 1.2. Kakšne so povezave med primarnimi elementi Kanban pristopa in petimi načeli vitkega razmišljanja, kot jih definirata Womack in Jones - vrednost z vidika kupca, vrednostni tok, vzpostavitev toka, izdelava na zahtevo, in stremljenje k popolnosti?

RQ 2. Kakšne so koristi in izzivi uporabe Kanban pristopa glede na obstoječo literaturo?

RQ 2.1. Kakšne so opažane koristi uporabe Kanban pristopa, ki jih navaja obstoječa literatura?

RQ 2.2. Kakšni so opažani izzivi uporabe Kanban pristopa, ki jih navaja obstoječa literatura?

RQ 3. Ali so vodila Kanban elementov, ki jih obstoječa literatura obravnava dobro določena in uporabna?

RQ 3.1. Ali so praktiki in strokovnjaki razlikovali med Kanban (velika začetnica) metodo in kanban (mala začetnica) sistemom?

RQ 4. Ali se v obstoječi literaturi za elemente Kanban pristopa le ponovno uporabljene agilni pojmi?

RQ 4.1. Ali se Scrum table pojmujejo kot Kanban table?

Prvo raziskovalno vprašanje se nanaša na primarne elemente Kanban. Drugo vprašanje preverja zaznane prednosti in izzive, ki jih prinaša Kanban. Tretje vprašanje raziskuje jasnost in uporabnost uveljavljenih vodil za vpeljavo Kanban v IT organizacije. Četrto vprašanje pa raziskuje, ali so se pojmi agilnosti le navidezno "preimenovali" v pojme Kanban.

3.3.2 Iskanje primarnih študij

Avtorja sta v sistematičnem pregledu zajela tako strokovno, kot tudi sivo literaturo, vključno s konferenčnimi zborniki, digitalnimi knjižnicami in disertacijami. Omejila sta se na literaturo, ki je bila objavljena po letu 1990, saj se ta letnica smatra kot začetek uporabe pojma vitkosti v razvoju programske opreme z objavo dela *The machine that changed the world* avtorjev Womack, Jones in Roos. Zgornja datumska meja objave literature pa je leto 2012. Iskanje se je izvajalo po povzetkih, ključnih besedah in naslovih.

Iskanje je potekalo elektronsko in tudi ročno, saj sta avtorja želela v pregled vključiti vse relevantne študije. Elektronsko iskanje je bilo izvedeno na naslednjih podatkovnih bazah: Scopus, Scirus, Google Scholar, ACM Digital Library, IEEE Xplore, Science Direct (Elsevier), Web of Science, InterScience (Wiley), Science.gov in InfoQ. Ročno iskanje je bilo izvedeno v revijah, kot so Empirical Software Engineering, The Journal of Systems and Software, in The Information and Software Technology Journal.

Med snovanjem protokola pregleda je bil določen seznam pojmov, ki se je nato uporabil kot temelj za definicijo iskalnih nizov. Ti so bili zgrajeni na osnovi strukturiranih raziskovalnih vprašanj po metodi PICOC. Sopomenke so bile povezane z logičnim operatorjem ALI, celoten iskalni niz pa je bil povezan z logičnimi operatorji IN. Iskalni nizi so predstavljeni v tabeli Tabela 2. Princip izgradnje iskalnega niza je bil enak ne glede na vrsto literature, vseeno pa se je iskalni niz razlikoval v elementu okvir (angl. Context), ki je bil za sistematične preglede literature drugačen, kot za monografsko literaturo.

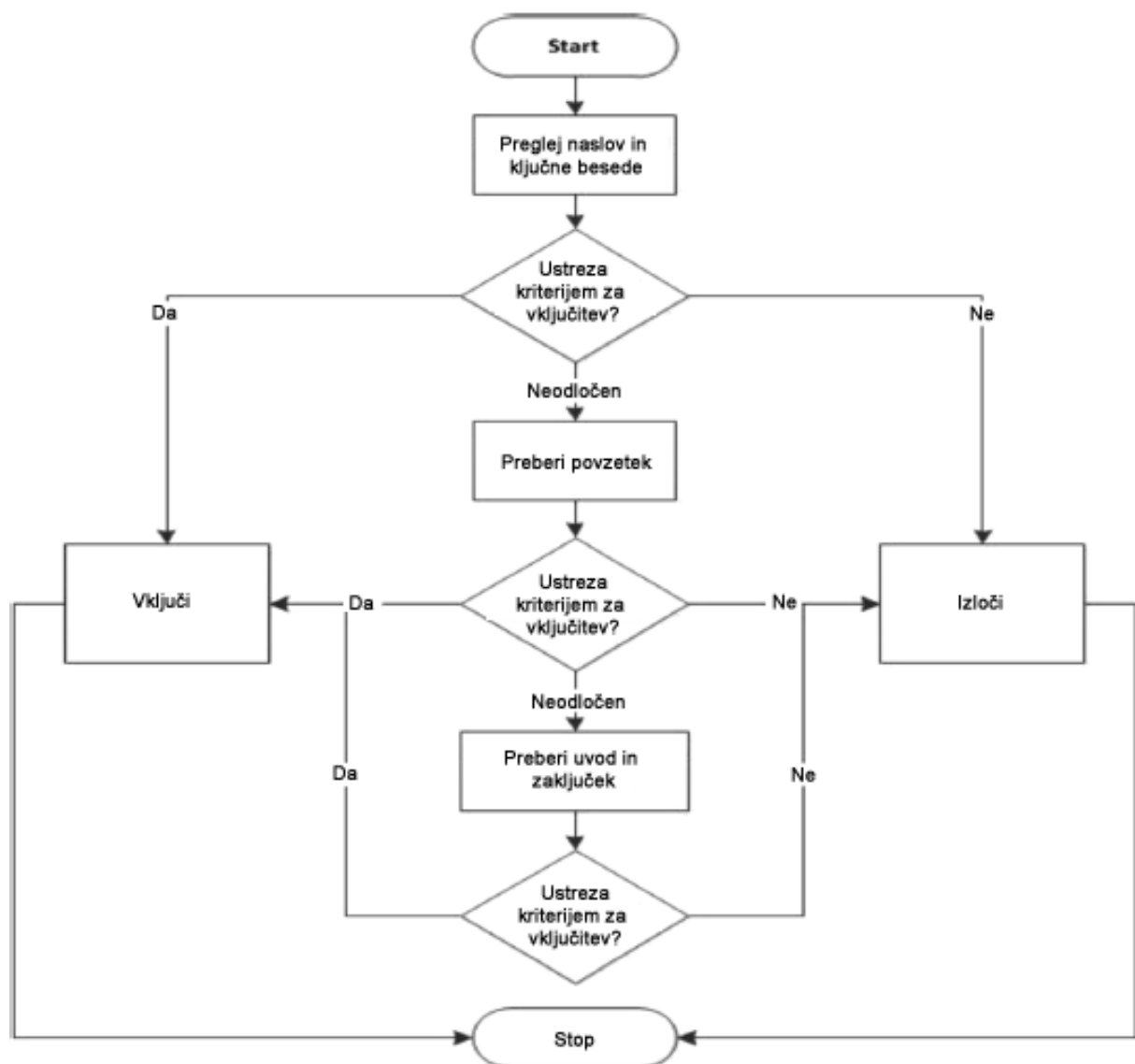
Element	Monografska literatura	Sistematični pregledi literature
Populacija	"Software engineering" OR "Software development" OR "Information technology" OR "Computer science" OR "IT project management" OR "Software project management" OR "Information	"Software engineering" OR "Software development" OR "Information technology" OR "Computer science" OR "IT project management" OR "Software project management" OR "Information technology project

	technology project management" OR "Software product development" OR "IT Product development"	management" OR "Software product development" OR "IT Product development"
Poseg	Kanban OR Lean OR "Continuous flow" OR "Pull system" OR "Lead time" OR "Cycle time" OR "Work in progress" OR "Work in process" OR WIP	Kanban OR Lean OR "Continuous flow" OR "Pull system" OR "Lead time" OR "Cycle time" OR "Work in progress" OR "Work in process" OR WIP
Primerjava	Agile OR Agility OR Scrum OR Leagile OR Scrumban	Agile OR Agility OR Scrum OR Leagile OR Scrumban
Izid	"Process improvement" OR "Cost reduction" OR "Reducing Cost" OR "Reduce cost" OR "High quality" OR "Higher quality" OR "Quality improvement" OR "Improving quality" OR "Improve quality"	"Process improvement" OR "Cost reduction" OR "Reducing Cost" OR "Reduce cost" OR "High quality" OR "Higher quality" OR "Quality improvement" OR "Improving quality" OR "Improve quality"
Okvir	Empirical OR Experiment OR "Evidence based" OR "Industrial setting" OR "Case study" OR "Action research"	"Literature Review" OR Overview OR "Research review" OR "Research synthesis" OR "Research integration" OR "Systematic review" OR "Integrative research review" OR "Integrative review"

Tabela 2: Iskalni nizi.

3.3.3 Izbira primarnih študij

Izbira primarnih študij je potekala v skladu s kriteriji za vključitev, ki so bili definirani v protokolu pregleda. Pregledovanje študij je potekalo v treh korakih. V prvem koraku sta avtorja preverila kriterije za vključitev nad naslovi in ključnimi besedami študij. Če odločitev za vključitev študije na podlagi teh podatkov ni bilo možno sprejeti, sta avtorja v drugem koraku kriterije preverila še nad povzetki. Če sta bila avtorja še vedno v dvomih, sta v tretjem koraku preverila še uvod in zaključek študije. Postopek izbire primarnih študij je z diagramom poteka prikazan na sliki Slika 8.



Slika 8: Postopek izbire primarnih študij.

Avtorja sta določila naslednje kriterije za vključitev študij v raziskavo:

- Študija naj bo napisana v angleškem jeziku.
- Študija naj bo objavljena med letoma 1990 in 2012.
- Študija naj vključuje empirične podatke iz industrijskega področja. Metodologija raziskave, ki je bila uporabljena v študiji – študije raziskovalne ali razlagalne narave (angl. exploratory or explanatory), ni pomembna.

- V študiji mora biti jasno razvidno, da se osredotoča na razvoj programske opreme z uporabo Kanban pristopa ali vitkega razvoja.
- Študija naj opisuje elemente in pristope, ki se uporabljajo za implementacijo Kanban.
- Če je bila študija objavljena večkrat, v več različnih virih, se v obravnavo vključi le nazadnje objavljena študija.

Kriterijev za izključitev študij avtorja nista posebej določala. Edini kriterij za izključitev je bil neskladnost s kriteriji za vključitev študije.

Vsaka od tako izbranih študij je bila nato preverjena še v postopku za ocenjevanje kakovosti študije. Postopek je bil zasnovan tako, da je bil primeren za preverjanje kakovosti ločeno vseh treh tipov literature, ki so nastopali v raziskavi: monografske literature, sistematičnih pregledov in sive literature. Raziskovalci so v ta namen izdelali orodje za preverjanje kakovosti.

		Quality Instrument		
Study Identifier (Unique ID)				
1	What is the study Type?	<input type="checkbox"/> ML	<input type="checkbox"/> SLR	<input type="checkbox"/> GL
<i>If the type is Monographic, complete section (A), if the type is SLR, complete section (B), otherwise, complete section (C)</i>				
Section (A): Monographic				
1	Does the study report any empirical research?	<input type="checkbox"/> Yes	<input type="checkbox"/> No	
2	Is there a clear description of the aim of the research?	<input type="checkbox"/> Yes	<input type="checkbox"/> No	<input type="checkbox"/> Partially
3	Is there a clear description of the study context?	<input type="checkbox"/> Yes	<input type="checkbox"/> No	<input type="checkbox"/> Partially
4	Is there an adequate justification for the research design?	<input type="checkbox"/> Yes	<input type="checkbox"/> No	<input type="checkbox"/> Partially
5	Is there an adequate description of the studied sample?	<input type="checkbox"/> Yes	<input type="checkbox"/> No	<input type="checkbox"/> Partially
6	Is there an adequate justification for the selected sample?	<input type="checkbox"/> Yes	<input type="checkbox"/> No	<input type="checkbox"/> Partially
7	Is there a clear description of the data collection?	<input type="checkbox"/> Yes	<input type="checkbox"/> No	<input type="checkbox"/> Partially
8	Is there a clear description of the data analysis?	<input type="checkbox"/> Yes	<input type="checkbox"/> No	<input type="checkbox"/> Partially
9	Are the findings of the study clearly stated?	<input type="checkbox"/> Yes	<input type="checkbox"/> No	<input type="checkbox"/> Partially
10	Is there sufficient data presented to support the findings?	<input type="checkbox"/> Yes	<input type="checkbox"/> No	<input type="checkbox"/> Partially
11	Are the research contribution and value adequately discussed?	<input type="checkbox"/> Yes	<input type="checkbox"/> No	<input type="checkbox"/> Partially
12	Are validity threats of the research adequately described?	<input type="checkbox"/> Yes	<input type="checkbox"/> No	<input type="checkbox"/> Partially
<i>If any of the answer is "No" for any of the first two questions, exclude the paper. The scoring procedure is Yes=2, Partial=1 and No or Unknown=0, calculate total from 4-12, if score is 12 or more, include. For more details about scoring, see "Ivarsson and Gorschek".</i>				
Section (B): Systematic Literature Review (SLR)				
1	Are the review's inclusion and exclusion criteria described and appropriate?	<input type="checkbox"/> Yes	<input type="checkbox"/> No	<input type="checkbox"/> Partially
2	Is the literature search likely to have covered all relevant studies?	<input type="checkbox"/> Yes	<input type="checkbox"/> No	<input type="checkbox"/> Partially
3	Did the reviewers assess the quality/validity of the included studies?	<input type="checkbox"/> Yes	<input type="checkbox"/> No	<input type="checkbox"/> Partially
4	Were the basic data/studies adequately described?	<input type="checkbox"/> Yes	<input type="checkbox"/> No	<input type="checkbox"/> Partially
<i>The scoring procedure is Yes=2, Partial=1 and No or Unknown=0, calculate total from 1-4, if score is 6 or more, include. For more details about scoring see "Kitchenham - Guidelines".</i>				
Section (C): Gray Literature Websites				
1	Is it an official website for Lean/Kanban/Agile/Software organizations/societies?	<input type="checkbox"/> Yes	<input type="checkbox"/> No	
2	Does it provide references for what has been written/described on the website?	<input type="checkbox"/> Yes	<input type="checkbox"/> No	
3	Is the data on the website quoted/cited from one or more of the included primary	<input type="checkbox"/> Yes	<input type="checkbox"/> No	
<i>If any of the answer is "No" for any of the first two questions or "Yes" for the third question, exclude the website.</i>				
Non-Websites (eBooks, Articles on InfQ)				
1	Does the study report any empirical research?	<input type="checkbox"/> Yes	<input type="checkbox"/> No	
2	Has the study been peer reviewed by an expert in the domain?	<input type="checkbox"/> Yes	<input type="checkbox"/> No	
3	Does the study have "Known" author?	<input type="checkbox"/> Yes	<input type="checkbox"/> No	
4	Does the author has h-index > 5? (using Google Scholar gadget)	<input type="checkbox"/> Yes	<input type="checkbox"/> No	
5	Does the study has been cited by > 5 studies? (using Google Scholar gadget)	<input type="checkbox"/> Yes	<input type="checkbox"/> No	
<i>If the answer is "No" for the 1st questions, exclude the study. If the answer is yes, proceed to question 2, if the answer is "Yes", include the study, if the answer is "No", proceed to question 3 and 4, if the answer is "No" for both of the questions, exclude the study.</i>				

Slika 9: Orodje za preverjanje kakovosti študij [6].

Ker sta sistematični pregled opravljala dva raziskovalca, je pri sprejemanju odločitev za vključitev primarnih študij med njima prihajalo do nesoglasij. V izogib konfliktom sta se raziskovalca pri izbiri primarnih študij držala algoritma, ki je prikazan v sliki Slika 8. Slika 8: Postopek izbire primarnih študij. Ujemanje med raziskovalcema pa je bilo merjeno z meritvijo Cohen Kappa. Strinjanje je predstavljala meritev Kappa, ki je bila po vrednosti višja od 0,75.

Kriteriji za vključitev so bili dvakrat preverjeni in dopolnjeni. Po prvem preverjanju se je ujemanje med raziskovalcema (angl. inter-rater reliability) povečala za 20%, po drugem, pa sta avtorja ugotovila, da so kriteriji ustrezno zanesljivi.

3.3.4 Postopek pridobivanja podatkov

Obrazec za pridobivanje podatkov je bil določen in preverjen v fazi načrtovanja pregleda med izdelavo dokumenta protokola pregleda.

Avtorja sta z namenom zagotavljanja zanesljivosti in doslednosti pridobivanja podatkov uporabila dve metodi: retestno metodo (angl. test-retest) in navzkrižno preverjanje (angl. cross checking). Po prvi sta avtorja ponovila postopek pridobivanja podatkov, po drugi pa sta navzkrižno preverila 12 naključno izbranih primarnih študij.

Avtorja sta pridobivanje podatkov, razen za primarne študije iz sive literature, izvedla hkrati z ocenjevanjem kakovosti primarnih študij, saj je bilo treba že pri tem pregledati celotno besedilo študij. Za primarne študije sive literature je postopek pridobivanja podatkov potekal povsem ločeno.

3.3.5 Sinteza podatkov

V sistematičnem pregledu sta avtorja opravila opisno sintezo podatkov z analitičnim pristopom, ki se uporablja za analizo študij v tekstovni obliki. Glede na raziskovalna vprašanja, ki sta jih avtorja zastavila v sistematičnem pregledu, sta kot najbolj primerno metodo za sintezo podatkov izbrala metodo *case survey* [18]. Metoda se uporablja za kategoriziranje in šifriranje sorodnih pojmov iz različnih kvalitativnih študij za namen kvantitativne analize. Podatki študij se povzamejo v skupni tabeli, pri čemer vrstice predstavljajo primarne študije, stolpci pa podatke, v tem primeru elemente Kanban pristopa. Za označevanje obravnavanja elementa v posamezni študiji se uporablja barvno kodiranje, ki je sestavljeno iz trobarvne sheme: zelena predstavlja pojme, ki so bili temeljito obravnavani v študiji, rumena predstavlja pojme, ki so bili le omenjeni, rdeča pa pojme, ki jih študija ni obravnavala.

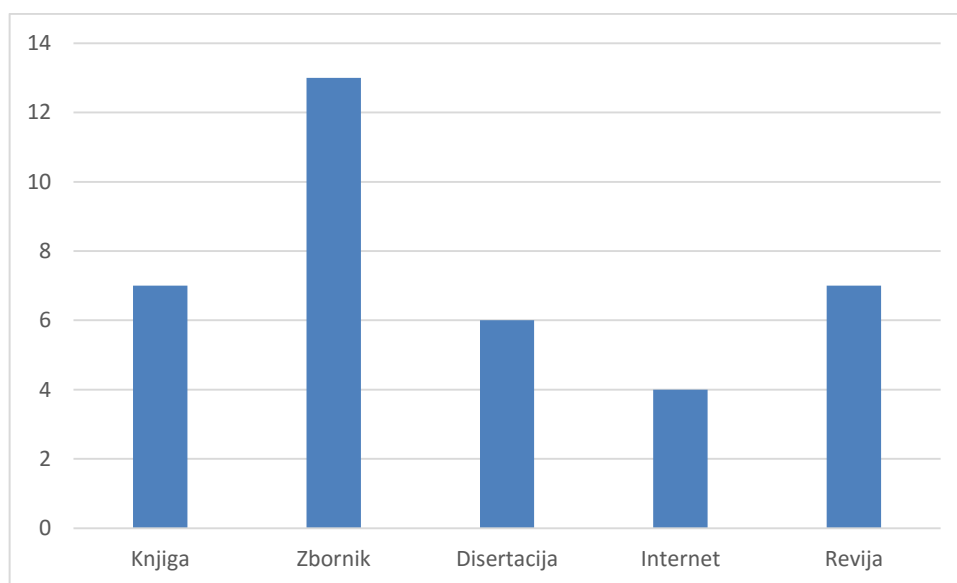
Kodirna shema je bila določena z uporabo vzajemne pretvorbe (angl. reciprocal translation), po tehniki metaetnografije (angl. Meta-ethnography), s katero se poskuša ključne pojme v vsaki izmed primarnih študij pretvoriti enega v drugega. Ko je bila kodirna shema določena, je bila posredovana raziskovalni komisiji (raziskovalci, ki so poznali področje raziskave), ki je shemo pregledala in podala pripombe. Glede na pripombe sta raziskovalca shemo dopolnila.

Končna verzija kodirne sheme je predstavljena v prilogi Priloga II – Kodirna shema.

3.3.6 Rezultati

3.3.6.1 Rezultati iskanja in ocenjevanja kakovosti literature

Iskanje literature je obrodilo skupno 7.809 študij. Po upoštevanju kriterijev za vključitev študij, sta raziskovalca v raziskavo vključila 54 primarnih študij. Po izvedbi ocenjevanja kakovosti študij, sta raziskovalca izločila dodatnih 17 študij, tako da je na seznamu primernih primarnih študij za raziskavo ostalo 37 študij. Razdelitev primarnih študij po vrstah gradiva je predstavljena na sliki Slika 10.



Slika 10: Porazdelitev primarnih študij glede na vrsto gradiva [6].

3.3.6.2 Rezultati raziskave

Primarne študije so bile razdeljene v štiri skupine glede na podobnost tipa študije in konteksta. Skupine so predstavljene v tabeli Tabela 3: Skupina glede na obravnavo Kanban v primarnih študijah.

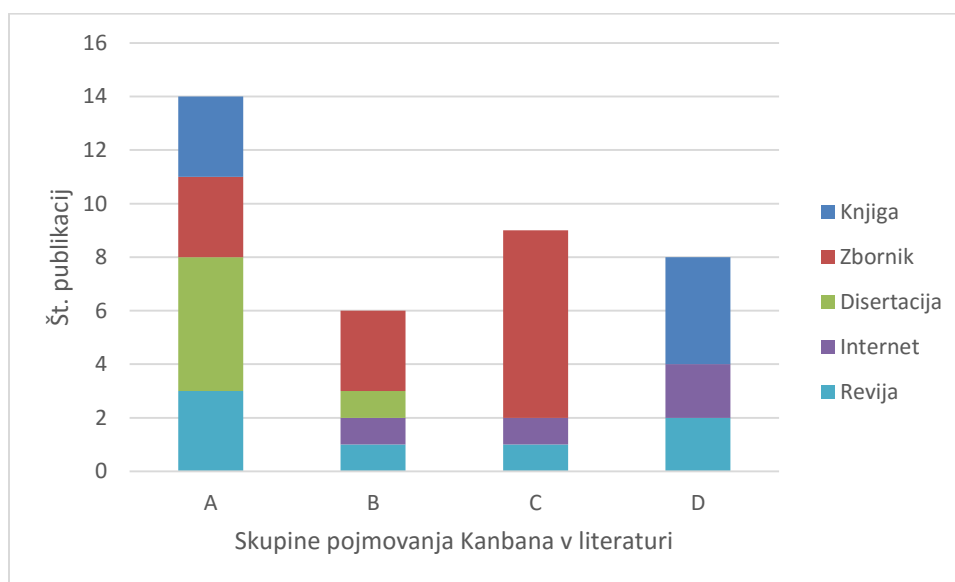
Tip	Skupina	Opis
A	Kanban kot element vitkosti	Skupina obravnava Kanban kot enega od osnovnih elementov vitkosti. Kanban velja za začetek prehoda od agilnosti k vitkosti.
B	Kanban kot agilna metodologija	Skupina obravnava Kanban kot agilno metodologijo. Kanban se uporablja za udejanjenje

manifesta agilnosti.

- | | | |
|----------|------------------------------|--|
| C | Hibrid med Kanban in Scrum | Skupina obravnava kombinacijo Kanban in Scrum. Kanban se uporablja za odpravljanje pomanjkljivosti in spodbujanje izrabe prednosti Scruma. |
| D | Kanban kot samostojno orodje | Skupina predstavlja Kanban kot samostojno orodje in ne kot element agilnosti ali vitkosti. |

Tabela 3: Skupina glede na obravnavo Kanban v primarnih študijah.

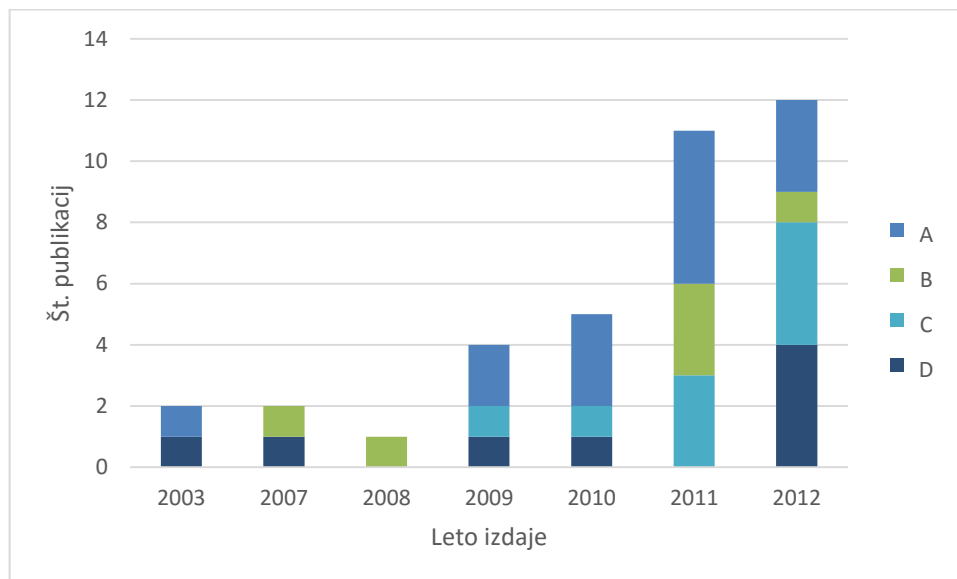
Slika Slika 11 predstavlja graf porazdelitve primarnih študij po skupinah po tipu študije. Na podlagi grafa lahko pridemo do več zaključkov. V disertacijah se navadno poskuša dokazati nek specifičen pojav. Iz grafa lahko razberemo, da se v disertacijah poskuša dokazati, da lahko pojem Kanbana kot samostojne metodologije obravnavamo tudi izven okvirjev agilnosti. Na podlagi tega lahko sklepamo, da Kanban kot samostojna metodologija še ni dobro uveljavljen. Na konferencah se vsebine nagibajo proti tipu C, kar pomeni, da se strokovnjaki in praktiki z vpeljevanjem Kanban ukvarjajo predvsem z izboljševanjem obstoječe agilne metodologije.



Slika 11: Porazdelitev primarnih študij po skupinah pojmovanja Kanban in po tipu študije.

Na sliki Slika 12 je prikazana porazdelitev števila primarnih študij po skupinah pojmovanja Kanban in po letu objave študije. Iz grafa lahko sklepamo, da je Kanban pristop razmeroma

nov pojem na področju IT, za njegovo uveljavljanje pa bo potrebno dodatno preučevanje in raziskovanje. Kanban se je na področju razvoja programske opreme pojavil leta 2003, večje zanimanje pa se je pokazalo šele po letu 2009.



Slika 12: Porazdelitev primarnih študij po skupinah pojmovanja Kanban in letu objave.

3.3.6.3 Elementi Kanbana

Avtorja sta v 37 primarnih študijah identificirala 20 konceptov, načel in tehnik povezanih s Kanbanom, ki sta jih poimenovala elementi Kanbana. V nadaljevanju so na slikah Slika 13, Slika 14, Slika 15 in Slika 16 prikazani podatki o obravnavi elementov Kanbana za vsako skupino primarnih študij posebej. Elementi Kanbana so označeni z oznakami C1 do C20, primarne študije pa z oznakami S1 do S37. Obravnava elementov Kanbana v primarni študiji je predstavljena z barvami: zelena predstavlja pojme, ki so bili temeljito obravnavani v študiji, rumena predstavlja pojme, ki so bili le omenjeni, rdeča pa pojme, ki jih študija ni obravnavala. Iz podatkov je razvidno, da so nekateri elementi v študijah obravnavani bolj pogosto in podrobno od drugih.

V nadaljevanju bomo opisali elemente Kanbana. Poleg opisa bomo podali tudi ugotovitve avtorjev pri obdelavi primarnih študij.

C1 Metoda Kanban, opisuje postopek vpeljave Kanban v organizacijo. Ugotovitve: postopek vpeljave naj bo postopen, metodologija naj bo preverjana in nenehno izboljševana.

C2 Kanban tabla je orodje za vizualizacijo delovnega toka. Na tabli so v stolpcih prikazane aktivnosti delovnega toka, ki morajo biti opravljeni od nastanka zahteve za izdelek oz.

storitev, pa vse do njegove predaje kupcu. Ugotovitve: začetna zasnova table je lahko določena na več načinov: nekateri strokovnjaki so predlagali uporabo obstoječih aktivnosti delovnega toka ob vpeljavi Kanban metode, drugi so predlagali uporabo aktivnosti klasičnega slapovnega modela (angl. waterfall model), tretji pa enostavnejšo možnost, ki vključuje tri osnovne aktivnosti, to so narediti, v delu in dokončani (angl. to do, in progress, done).

C3 Izvlečni sistem (angl. pull system) pristop, po katerem se proces razvoja začne takrat, ko kupec poda zahtevo. Ugotovitve: v zvezi z izvlečnim sistemom so strokovnjaki ugotavljali, da le-ta zahteva, da se določi omejitev števila delovnih nalog, ki so sočasno v delu, oz. dela v teku (angl. work-in-progress (WIP) limit). Nova delovna naloga se lahko začne izvajati le, če omejitev še ni bila dosežena. Poleg tega je zaželeno, da se za izvajanje izbere delovna naloga z najvišjo prioriteto.

C4 Razvrščanje po prioriteti (angl. prioritized queue) se nanaša na seznam delovnih nalog, ki jih je treba opraviti, razvrščenih po prioriteti izvedbe. Ugotovitve: strokovnjaki so razglabljali o načinih določanja prioritete, med katerimi so navedli kriterije: glede na vrednost, nujnost in pomembnost delovne naloge, glede na stroške, ki nastanejo zaradi zamude dokončanja naloge ter glede na kadrovske razpoložljivost.

C5 Kriteriji vključitve (angl. inclusion criteria) je eden izmed pomembnejših elementov metodologije Kanban. S kriteriji se preverja, ali bo imela delovna naloga za kupca dodano vrednost in bo tako primerna za izvedbo. Če to velja, se lahko delovno nalogo doda na Kanban tablo. Ugotovitve: v predhodni študiji je bilo ugotovljeno, da so bili le v eni od primarnih študij navedeni kriteriji vključitve.

ID	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14	C15	C16	C17	C18	C19	C20
S5	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
S6	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
S7	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
S14	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
S18	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
S21	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
S22	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
S24	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
S25	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
S26	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
S27	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
S29	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
S31	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
S34	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●

Slika 13: Obravnava elementov Kanbana v primarnih študijah tipa A [6].

C6 Omejitev obsega dela (angl. work-in-progress (WIP)) je največje dovoljeno število sočasno izvajanih delovnih nalog v posamezni aktivnosti delovnega toka. WIP je ena izmed pomembnejših lastnosti Kanban pristopa in njegova vpeljava glavni cilj. Ugotovitve: točne definicije avtorji študij niso podali. Večina študij je predlagala, da organizacija določi WIP meje tako, da s poskušanjem ugotovi za dane pogoje najboljše vrednosti.

C7 Zaključena delovna naloga (angl. done item). Element se nanaša na definicijo kriterijev, po katerih se delovna naloga smatra kot zaključena. Ugotovitve: kljub temu, da se ta element smatra za enega izmed ključnih pri vzpostavljanju zveznega toka (angl. continuous flow), ustrezne definicije oz. obravnave v študijah ni bilo. Nekatere študije so sicer trdile, da so določile zaključeno delovno nalogo, a razlage, kako je bila določena, ne navajajo. V eni študiji so zaključeno delovno nalogo definirali s kontrolnim seznamom, ki je temeljil na smernicah in standardih kakovosti pisanja programske kode.

C8 Vzratno pomikajoča se delovna naloga (angl. reverse item) je delovna naloga, ki se ko Kanban tabli pomika v vzratni smeri. Primer take naloge je odpravljanje napake, ki zahteva nadaljnji razvoj. Ugotovitve: strokovnjaki posebnega poudarka temu elementu niso dajali. Predlagali so dve rešitvi, in sicer krožni delovni tok ter ponovno postavitve naloge na seznam zahtev izdelka (angl. backlog).

C9 Validirano učenje (angl. validated learning) je proces učenja, pri katerem se znanje pridobiva tako, da se idejna rešitev implementira, nato pa se njen učinek ovrednoti s poslovnega vidika. Ta postopek se nato ponavlja, pri tem pa nastaja novo znanje. Ugotovitve: strokovnjaki so ugotavljali, da je nadziranje in učenje eden od ključnih pokazateljev učinkovitosti s katerimi merimo uspešnost projekta.

C10 Čas cikla / potrební čas (angl. cycle time/lead time) je ključni element, ki določa učinkovitost in zmogljivost delovnega procesa, saj se z njim meri učinke Kanbana. Ugotovitve: strokovnjaki si v definiciji elementa niso bili enotni. Nekateri so celo zamenjevali pojma čas cikla in potrební čas. Drugi so čas cikla definirali kot čas, ki preteče od začetka do zaključka dela na funkcionalnosti, spet tretji pa so čas cikla določili kot čas med predajami izdelkov.

ID	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14	C15	C16	C17	C18	C19	C20
S2	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
S3	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
S9	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
S28	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
S30	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
S32	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●

Slika 14: Obravnava elementov Kanbana v primarnih študijah tipa B [6].

C11 Orodje za merjenje performas. Element opisuje na kakšen način so izmerjeni učinki Kanban. Ugotovitve: v nekaterih študijah je bila kot orodje za merjenje učinkovitosti navedena uporaba kumulativnega diagrama poteka (angl. cumulative flow diagram, CDF) osnovanega na času cikla in WIP. Druge študije so spodbujale dnevni nadzor napredovanja z uporabo diagrama preostalega dela (angl. Burn Down Chart).

C12 Ozko grlo (angl. bottleneck) je tista točka v delovnem procesu, na kateri se delovne naloge iz predhodne točke prevzemajo počasneje, kot se v predhodni točki zaključujejo. Ugotovitve: v študijah so ugotavljali, da se ozka grla lahko odkrijejo z vizualizacijo delovnega toka, ali z vpeljavo orodja za odkrivanje ozkih grl, kot to predlaga teorija omejitev. Predlagali so tudi razdelitev ozkih grl v več delov ter postopno vrednotenje njihovih vrednosti.

C13 Čakalna vrsta (angl. slack or buffer). Potrebni čas (angl. lead time) lahko obravnavamo kot vsoto časa, ko je delovna naloga v delu, in časa, ko naloga čaka na njeno izvajanje (nihče ne opravlja dela na nalogi). Za naloge, ki so v čakanju, lahko uvedemo čakalno vrsto. Ugotovitve: strokovnjaki so v študijah navajali, da vpeljava čakalne vrste omogoča merjenje časa, ki ga delovne naloge "preživijo" v čakanju, obvladovanje nihanja časa cikla procesa ter preprečevanju oz. zmanjšanju števila ozkih grl. Ugotavljali so tudi, da je treba velikost čakalne vrste periodično preverjati in jo po potrebi spreminjati.

C14 Odkrivanje odpadkov. Kot odpadki obravnavamo vse aktivnosti, ki ne doprinašajo nobene dodane vrednosti za kupca. Take aktivnosti je zato treba odstraniti. Ugotovitve: strokovnjaki so soglašali v ugotovitvi, da vpeljava Kanban pripomore k vizualni komunikaciji in odkrivanju odvečnih aktivnosti, s tem pa k hitremu odzivu na kupčeve potrebe.

C15 Sodelovanje v ekipi. Med člani ekipe velikokrat pride do pomanjkanja sodelovanja in povezanosti. Ugotovitve: avtorji študij so kot problem izpostavljali razkorak v komunikaciji med tehnično podkovanimi razvijalci programske opreme na eni, in ostalimi vpletenimi v

poslovni proces na drugi strani. Trdili so, da se je z vpeljavo Kanbana nemudoma izboljšala komunikacija med člani ekipe, ter med ekipo in vodstvom organizacije.

ID	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14	C15	C16	C17	C18	C19	C20
S4	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
S8	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
S10	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
S11	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
S12	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
S15	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
S17	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
S33	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
S37	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●

Slika 15: Obravnava elementov Kanbana v primarnih študijah tipa C [6].

C16 Sestanki. Ugotovitve: nekateri strokovnjaki so sestanke obravnavali kot odpadke, in bi jih bilo zato treba odpraviti. Drugi pa so spodbujali prirejanje kratkih dnevnih posvetov, na katerih se obravnavajo tekoče problematike. Spet tretji predlagajo, da morajo imeti ekipe svobodo pri izbiri dolžine in pogostosti sestankov.

C17 Avatar je vizualna predstavitev osebe, ki je vršilec delovne naloge. Takšen prikaz ponuja pomembne informacije pri sprejemanju odločitev. Ugotovitve: strokovnjaki so navedli, da podatki na Kanban tabli pripomorejo k sprejemanju objektivnih odločitev. Vodstvu je na voljo preprost pregled podatkov o kadrovske zasedenosti, ki jim je v pomoč pri razporejanju kadrov.

C18 Planiranje in ocenjevanje. Ugotovitve: več strokovnjakov je planiranje spoznalo kot odpadke in predlagalo zmanjšanje njenega obsega, s čimer bi dosegli povečanje učinkovitosti zaposlenih in zadovoljstva kupcev. Poudarjali so, da se z zmanjšanjem časa, porabljenega za planiranje, zmanjša tudi potreben čas (angl. lead time). Vseeno pa planiranju in retrospektivnim aktivnostim dajejo pomen, saj s povratno informacijo članov ekipe omogočajo izboljševanje procesa.

C19 Politika Kanban pristopa so smernice, ki določajo kaj je treba narediti in na kakšen način. Ugotovitve: Strokovnjaki so navajali, da se z upoštevanjem politike lažje dosegajo dogovori in zmanjšujejo spori, ter se v razpravah zmanjšuje pristranskost. Poleg tega so predlagali, da člani ekipe sodelujejo pri oblikovanju Kanban politike.

C20 Povratna informacija (angl. feedback loop). Glavni cilj majhnih in pogostih izdaj programske opreme je pridobivanje povratne informacije s strani uporabnikov. Ugotovitve:

Več strokovnjakov je poudarilo pomen povratne informacije. Povratno informacijo lahko obravnavamo kot dejstvo in je nedvomno bolj uporabna, kot pa naša ugibanja o potrebah uporabnikov. Kljub temu avtorji niso razglabljali, kako bi lahko Kanban pripomogel k pridobivanju povratne informacije.

ID	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14	C15	C16	C17	C18	C19	C20
S1	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
S13	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
S16	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
S19	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
S20	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
S23	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
S35	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
S36	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●

Slika 16: Obravnava elementov Kanbana v primarnih študijah tipa D [6].

3.3.6.4 Koristi in izzivi uporabe Kanban pristopa

Avtorja sta zbrala seznama koristi, ki jih prinaša vpeljava Kanban pristopa, pa tudi izzive, kot jih navajajo raziskovalci v primarnih študijah. Korist ali izziv sta na seznam dodala le, če je bil naveden v vsaj petih primarnih študijah. Večina primarnih študij je navajala koristi in izzive vezane na upravljanje procesa razvoja programske opreme in sam proces razvoja programske opreme. Seznama sta predstavljena v tabelah Tabela 4 in Tabela 5. V prvem stolpcu so navedene koristi oz. izzivi vpeljave Kanban pristopa v organizacijo, v drugem stolpcu pa odstotek študij, v katerih se koristi oz. izzivi navajajo.

Koristi vpeljave Kanban pristopa

Odstotek primarnih študij v %

Izboljšuje vizualni nadzor in s tem podpira in olajša sprejemanje odločitev	45,9
Olajša usklajevanje večfunkcijskega ekipnega dela in uvaja samoorganizacijo	37,8
Spodbuja vpeljavo ciklov kakovosti in opravil, ki zagotavljajo kaizen (postopek nenehnega izboljševanja)	29,7
Zmanjšuje čas cikla, oz. potrebni čas	29,7
Povečuje zadovoljstvo kupca in omogoča ustvarjanje izdelkov visokega	27,0

cenovnega razreda

Zmanjšuje tehnična tveganja in tveganja, ki jih prinaša tržišče	24,3
Spodbuja razvoj strategij nenehnega izboljševanja	45,9
Povečuje predvidljivost izdaje novih izdelkov kupcu z upoštevanjem spreminjanja kupčevih zahtev	35,1
Zagotavlja razvoj znanja in spodbuja povezanost članov ekip	16,2
Usmerja in olajšuje upravljanje z organizacijskimi spremembami in spremembami organizacijske kulture	32,4
Povečuje kakovost izdelkov, kar se kaže v zmanjšanju števila okvar, povečanju števila izdelkov, ki ustrezajo kakovostnim merilom, in zmanjšanju števila napak.	16,2

Tabela 4: Seznam koristi vpeljave Kanban pristopa v organizacijo.

Izzivi pri vpeljavi Kanban pristopa	Odstotek primarnih študij v %
Težko je določiti ustrezne metrike. Na primer potrebni čas (angl. lead time) se je za merjenje performans izkazal kot težaven	27,0
Vpeljava Kanban pristopa zahteva temeljito poznavanje konceptov, načel in praks vitkosti	21,6
Uporabljajo se nejasne definicije osnovnih elementov Kanban, kot na primer definicija zaključene delovne naloge	27,0
Ni ustreznih smernic za vpeljavo pristopa in elementov Kanban.	54,4
IT organizacije s težavo vpeljujejo nekatere elemente Kanban pristopa, kot je npr. vzpostavljanje zveznega toka. (angl. flow)	27,0
Treba je spremeniti organizacijo in strukturo ekip razvijalcev	16,2

Vpeljava Kanban pristopa v proces razvoja programske opreme in njegova integracija z obstoječimi agilnimi tehnikami je zapletena, draga in časovno potratna 13,5

Tabela 5: Seznam izzivov pri vpeljavi Kanban pristopa v organizacijo.

3.3.6.5 Jasnost obstoječih smernic za vpeljavo Kanban pristopa v organizacijo

Naj na tem mestu ponovno omenimo Andersonovo razlikovanje med Kanban metodo in kanban sistemom, kot izvlečnim sistemom [2]. Analiza primarnih študij je pokazala, da so praktiki in strokovnjaki ob obstoječih, pomanjkljivo določenih smernicah Kanban, upravičeno v dvomih. V zvezi s tem članek [6] opozarja na zmešnjavo pri uporabi besede kanban z malo začetnico in Kanban z veliko začetnico.

Le avtorji dveh primarnih študij so prevzeli Andersonovo definicijo z razlikovanjem med pojmom Kanban, zapisanim z malo in veliko začetnico. Avtorji drugih dveh študij so pojma v študiji med sabo mešali. V nekaterih študijah pa so Kanban definirali drugače, npr. kot orodje učenja, s katerim se postopoma vpeljuje elemente vitkosti. Kanban se je v študijah navajal tudi kot orodje za spreminjanje Scruma, tako da se z njim ukinja iterativni razvoj in prikaz preostalega dela (angl. burndown chart), ter vpeljuje vrste z določenimi prioritetami. Ena primarna študija je uporabljala povsem lastno pojmovanje, s katerim je besedo kanban uporabila za kanban v industrijskem kontekstu, besedo Kanban pa za Kanban v kontekstu razvoja programske opreme.

Tudi pri uporabi pojma kanban, kot izvlečnega sistema avtorji primarnih študij niso bili konsistentni. Nekateri so v začetku študije pravilno definirali kanban kot izvlečni sistem, ki omogoča zvezni delovni tok, kasneje pa so pojem očitno uporabljali tudi za razlago Kanban metode. Skoraj polovica primarnih študij sploh ni uporabljala pojma kanban (kot izvlečni sistem).

V eni izmed študij so Kanban interpretirali kot samostojno metodologijo, ki je po vpeljavi v organizacijo povsem zamenjala Scrum. Tako pojmovanje Kanban metode je seveda v nasprotju z Andersonovo definicijo, ki Kanban ne pojmuje, kot metodologijo za upravljanje delovnega procesa, ali razvoja programske opreme.

Približno četrtnina primarnih študij se je strinjala z elementi Kanban: 1) Vizualizacija delovnega toka (angl. Visualize Workflow), 2) Omejitev obsega dela v teku (angl. Limit Work-in-Progress) in 3) Merjenje in upravljanje delovnega toka (angl. Measure and Manage

Flow. Poleg teh sta dve študiji definirali dodatna elementa, ki v zaporedju nastopita pred omenjenimi elementi: 1) začni (vpeljavo Kanban) tako, da počneš natančno to, kar si počel doslej, in 2) izdelaj skico delovnega toka (angl. map the value stream). Dve študiji sta Kanban predstavili kot agilno metodologijo, kar je ponovno v nasprotju z Andersonovo definicijo. Ena študija je opisala nagel prehod od Scruma v Kanban. Tudi ta primer se zato ne sklada z Andersonovo definicijo, ki predvideva postopno izboljševanje.

3.3.6.6 Kanban tabla

Analiza primarnih študij je pokazala, da je pojem in način uporabe Kanban table slabo raziskan in dokumentiran. Avtorji primarnih študij se tako sprašujejo kakšna je sploh razlika med Kanban in Scrum tablo, pri čemer ugotavljajo, da je to »tista mala številka 2 pri imenu stolpca na Kanban tabli«. S tem seveda mislijo na omejitev dela v teku. Drugi uporabo Kanban table primerjajo s Scrum sprinti, ki so časovno omejeni, pri čemer časovni okvir ustreza omejitvi dela v teku.

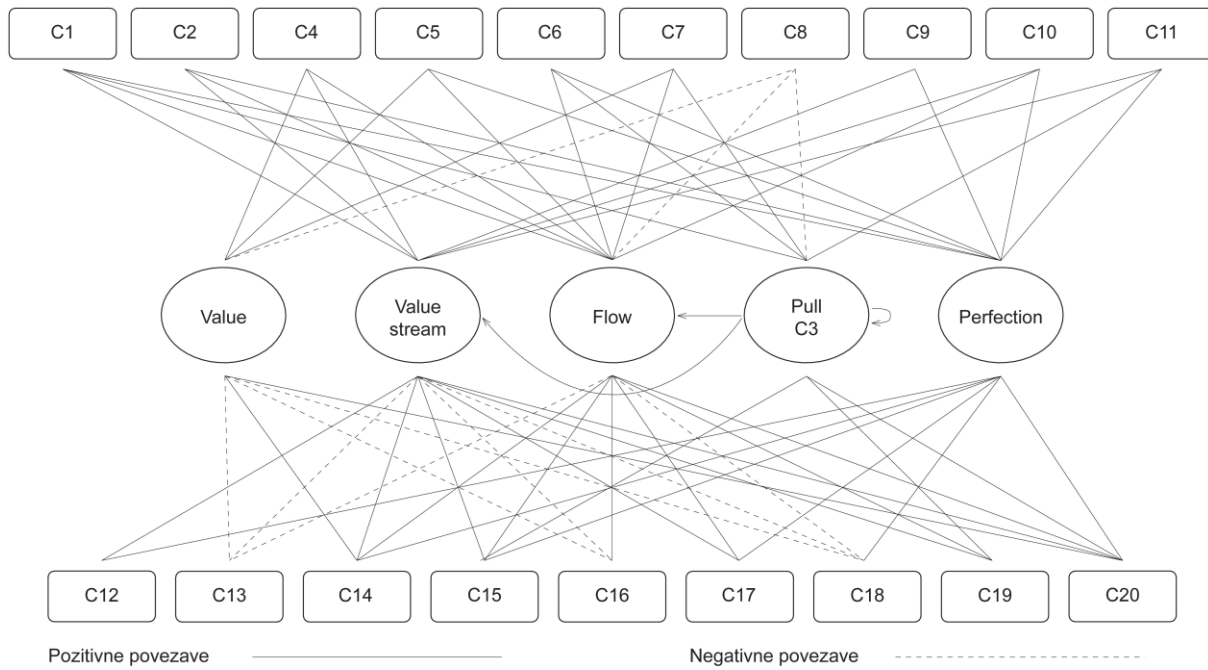
Nekateri avtorji trdijo, da je končna oblika Kanban table nastala na podlagi prilagoditev Scrum table, katere konkretne prilagoditve so bile narejene, pa ne navajajo. Spet drugi navajajo, da lahko Kanban tabla nudi tudi pregled nad težavami, izzivi in ozkimi grli. Na kakšen način se lahko to doseže, niso navedli.

3.3.6.7 Povezave med elementi Kanbana in načeli vitkosti

V nobeni od primarnih študij se ni obravnavalo povezav med elementi Kanban in elementi vitkosti ali manifestom agilnosti. V želji, da bi zmanjšala informacijsko vrzel, ki je tako nastala, sta avtorja poskušala skupaj obravnavati elemente Kanban pristopa in elemente vitkosti.

Povezave so bile določene na podlagi rezultatov tematske analize, ki sta jo avtorja opravljala za določitev kodirne sheme. Odkrila sta dve vrsti povezav: pozitivne in negativne. Negativne povezave so tiste, ki so bile v vsaj eni primarni študiji navedene kot negativne. Slika Slika 17 prikazuje povezave med elementi Kanban pristopa in načeli vitkosti.

Avtorja sta ugotovila, da bo implementacija katerega koli od naslednjih elementov Kanbana: C8, C13, C16, ali C18, ob vpeljavi Kanban pristopa negativno učinkovala na štiri od petih načel vitkosti: določitev vrednost z vidika kupca (angl. specify value), opredelitev načrta vrednostnega toka (angl. identify the value stream), vzpostavitev toka z izločanjem odpadkov (angl. make the value-creating steps flow), izdelava na zahtevo (angl. pull).



Slika 17: Povezave med elementi Kanban pristopa in načeli vitkosti [6].

Poglavje 4 Izvedba sistematičnega pregleda literature

4.1 Uvod

V tem poglavju bomo predstavili sistematični pregled literature, ki je bil izveden kot nadaljevanje oz. nadgradnja pregleda v [6]. Elementi protokola pregleda so bili povzeti po tem sistematičnem pregledu in nato prilagojeni tako, da ustrezajo obsegu pričujočega dela, ki je nekoliko spremenjen – sistematični pregled smo omejili le na znanstveno literaturo.

Ker sistematični pregled temelji na [6], se bomo tudi v tem delu držali smernic, ki sta jih Kitchenham in Charters predstavila v [7], poleg tega pa upoštevali odločitve v zvezi z izvedbo postopka pregleda, ki sta jih sprejela avtorja v [6].

4.2 Načrtovanje

V nadaljevanju bomo predstavili protokol pregleda in njegove elemente, ter obrazec za pridobivanje podatkov iz primarnih študij. Elementi protokola pregleda, ki so bili v [6] ustrezno dokumentirani, so bili uporabljeni tudi v naši raziskavi. Ker izvorni protokol pregleda iz [6] ni bil objavljen v celoti v študiji in ker ga v času nastanka tega dela ni bilo mogoče pridobiti, smo nekatere elemente protokola morali določiti sami. Ti so v nadaljevanju ustrezno dokumentirani in obrazloženi.

Enako velja tudi za obrazec za pridobivanje podatkov iz primarnih študij. Tudi ta ni bil objavljen v študiji [6]. Ker smo želeli ohraniti podobno strukturo pregleda in poročanja rezultatov, kot v [6], smo obrazec za pridobivanje podatkov sestavili sami na podlagi predstavljenih rezultatov sistematičnega pregleda [6].

4.2.1 Protokol pregleda

4.2.1.1 Ozadje in cilji sistematičnega pregleda

Kot smo v uvodnem poglavju navedli razširjenost uporabe Kanban metode na področju razvoja programske opreme iz leta v leto narašča. Sistematični pregled, ki sta ga izvedla avtorja v [6] je prikazal stanje obravnave Kanban pristopa v literaturi do leta 2012. S pričujočim delom smo želeli ta pregled nadgraditi tako, da smo raziskali kako se obravnavanje Kanban pristopa v literaturi pojavlja po letu 2012, ter kakšne so spremembe glede na stanje v letu 2012. Skratka, želeli smo preveriti kakšen vpliv je imel sistematični pregled [6] na dojemanje, razlaganje in raziskovanje Kanbana po njegovi objavi.

Cilji pregleda ostajajo enaki tistim iz [6], torej: razjasniti vodila in elemente Kanban pristopa, da bi spodbudili uspešno vpeljavo Kanban pristopa na področje razvoja programske opreme z jasnim definiranjem osnovnih pojmov Kanban pristopa in predstavitevjo prednosti in slabosti uporabe Kanban pristopa v IT organizacijah. K tem pa smo dodali še enega: primerjati trenutno stanje obravnave Kanban pristopa v literaturi z ugotovitvami sistematičnega pregleda [6].

4.2.1.2 Raziskovalna vprašanja

Ker smo želeli ohraniti enako strukturo raziskave kot v [6], kar nam omogoča lažjo primerjavo rezultatov, smo raziskovalna vprašanja prevzeli iz [6].

4.2.1.3 Strategija iskanja primarnih študij

Iskanje primarnih študij je potekalo na virih, kot so navedeni v [6]. Ker je pričujoč sistematični pregled usmerjen v odkrivanje izključno strokovne literature smo iz seznama virov odstranili tiste, ki nudijo predvsem sivo literaturo. Končni seznam virov primarnih študij je tako naslednji:

- Scopus
- ACM Digital Library
- IEEE Xplore
- Science Direct
- Web of Science
- InterScience (Wiley)

Avtorja v [6] navajata, da je bilo izvedeno tudi ročno iskanje raznih revij, med katerimi omenjata "Empirical Software Engineering", "The Journal of Systems and Software" in "The Information and Software Technology Journal". Podroben seznam virov sta avtorja zapisala v protokol pregleda, do katerega pa kot že rečeno pri izvedbi raziskave nismo imeli dostopa. Zaradi tega smo se pri ročnem iskanju revij omejili na zgoraj navedene. Članki iz revij "The Journal of Systems and Software" in "The Information and Software Technology Journal" so objavljeni v ostalih virih, ki smo jih našli zgoraj. Tako je bilo treba ročno pregledati le revijo "Empirical Software Engineering".

Točni iskalni nizi in vse podrobnosti poteka iskanja so navedene v prilogi Priloga III – Seznam virov literature s podrobnostmi izvedbe iskanja.

4.2.1.4 Kriteriji za vključitev in izključitev primarnih študij in postopek za vključitev študije

Kriteriji za vključitev primarnih študij v obravnavo so bili prevzeti iz [6]. Kriterij za omejevanje po letu objave študije smo prilagodili tako, da se upošteva literatura, ki je bila objavljena od vključno 2013 dalje. Končni seznam kriterijev za vključitev je tako naslednji:

- Študija naj bo napisana v angleškem jeziku.
- Študija naj bo objavljena po letu 2012.
- Študija naj vključuje empirične podatke iz industrijskega področja. Metodologija raziskave, ki je bila uporabljena v študiji – študije raziskovalne ali razlagalne narave (angl. exploratory or explanatory), ni pomembna.
- V študiji mora biti jasno razvidno, da se osredotoča na razvoj programske opreme z uporabo Kanban pristopa ali vitkega razvoja.
- Študija naj opisuje elemente in pristope, ki se uporabljajo za implementacijo Kanban.
- Če je bila študija objavljena večkrat, v več različnih virih, se v obravnavo vključi le zadnje objavljena študija.

Kriteriji za izključitev študij niso bili posebej določeni, zato edini kriterij za izključitev ostane neskladnost s kriteriji za vključitev.

Postopek za vključitev študij se je prevzel iz [6], prikazuje ga slika Slika 8.

4.2.1.5 Orodje za ocenjevanje kakovosti študije

Orodje za ocenjevanje kakovosti študije je bilo prevzeto iz [6]. Ker smo se v tem sistematičnem pregledu omejili le na strokovno literaturo, smo iz orodja za ocenjevanje kakovosti izključili dele, ki se nanašajo na sivo literaturo. Končna verzija orodja za ocenjevanje kakovosti študije je prikazana na sliki Slika 18.

Study Identifier (Unique ID)		Quality Instrument		
1	What is the study Type?	<input type="checkbox"/> ML	<input type="checkbox"/> SLR	
<i>If the type is Monographic, complete section (A), if the type is SLR, complete section (B), otherwise, complete section (C).</i>				
Section (A): Monographic				
1	Does the study report any empirical research?	<input type="checkbox"/> Yes	<input type="checkbox"/> No	
2	Is there a clear description of the aim of the research?	<input type="checkbox"/> Yes	<input type="checkbox"/> No	<input type="checkbox"/> Partially
3	Is there a clear description of the study context?	<input type="checkbox"/> Yes	<input type="checkbox"/> No	<input type="checkbox"/> Partially
4	Is there an adequate justification for the research design?	<input type="checkbox"/> Yes	<input type="checkbox"/> No	<input type="checkbox"/> Partially
5	Is there an adequate description of the studied sample?	<input type="checkbox"/> Yes	<input type="checkbox"/> No	<input type="checkbox"/> Partially
6	Is there an adequate justification for the selected sample?	<input type="checkbox"/> Yes	<input type="checkbox"/> No	<input type="checkbox"/> Partially
7	Is there a clear description of the data collection?	<input type="checkbox"/> Yes	<input type="checkbox"/> No	<input type="checkbox"/> Partially
8	Is there a clear description of the data analysis?	<input type="checkbox"/> Yes	<input type="checkbox"/> No	<input type="checkbox"/> Partially
9	Are the findings of the study clearly stated?	<input type="checkbox"/> Yes	<input type="checkbox"/> No	<input type="checkbox"/> Partially
10	Is there sufficient data presented to support the findings?	<input type="checkbox"/> Yes	<input type="checkbox"/> No	<input type="checkbox"/> Partially
11	Are the research contribution and value adequately discussed?	<input type="checkbox"/> Yes	<input type="checkbox"/> No	<input type="checkbox"/> Partially
12	Are validity threats of the research adequately described?	<input type="checkbox"/> Yes	<input type="checkbox"/> No	<input type="checkbox"/> Partially
<i>If any of the answer is "No" for any of the first two questions, exclude the paper. The scoring procedure is Yes=2, Partial=1 and No or Unknown=0, calculate total from 4-12, if score is 12 or more, include. For more details about scoring, see "Ivarsson and Gorschek".</i>				
Section (B): Systematic Literature Review (SLR)				
1	Are the review's inclusion and exclusion criteria described and appropriate?	<input type="checkbox"/> Yes	<input type="checkbox"/> No	<input type="checkbox"/> Partially
2	Is the literature search likely to have covered all relevant studies?	<input type="checkbox"/> Yes	<input type="checkbox"/> No	<input type="checkbox"/> Partially
3	Did the reviewers assess the quality/validity of the included studies?	<input type="checkbox"/> Yes	<input type="checkbox"/> No	<input type="checkbox"/> Partially
4	Were the basic data/studies adequately described?	<input type="checkbox"/> Yes	<input type="checkbox"/> No	<input type="checkbox"/> Partially
<i>The scoring procedure is Yes=2, Partial=1 and No or Unknown=0, calculate total from 1-4, if score is 6 or more, include. For more details about scoring see "Kitchenham - Guidelines".</i>				

Slika 18: Orodje za ocenjevanje kakovosti primarnih študij v sistemaičnem pregledu literature.

Orodje za ocenjevanje kakovosti je že bilo preverjeno v okviru študije [6], zato smo ocenili, da ponovno preverjanje ustreznosti kriterijev orodja ni potrebno. Ocenjevanje kakovosti študij je opravljal le avtor pričujočega dela, zato ujetja odločitev ni bilo treba ocenjevati. Orodje se je uporabilo v namen olajšanja sprejema odločitve za vključitev ali izključitev primarne študije, torej kot razširitev kriterijev za vključitev študije.

Ocenjevanje kakovosti študije je potekalo kot samostojni postopek, ločen od postopka pridobivanja podatkov iz primarnih študij.

4.2.1.6 Strategija pridobivanja podatkov

Ker smo želeli primerjati rezultate postopka pridobivanja podatkov s tistimi v [6], bi bilo idealno, če bi za pridobivanje podatkov uporabili isti obrazec, kot avtorja v [6], vendar je bil ta izdelan v okviru protokola pregleda, ki ob izvedbi pričujočega dela ni bil na voljo. Zaradi tega je bil obrazec za pridobivanje podatkov izdelan na podlagi rezultatov sistematičnega pregleda v [6].

Kot smo že omenili je bil postopek pridobivanja podatkov popolnoma ločen od postopka ocenjevanja kakovosti primarnih študij.

4.2.1.7 Sinteza podatkov

Po zgledu postopka sinteze podatkov v [6] se je tudi v tem delu uporabilo metodo *case survey* [18]. Za označevanje obravnavanja elementa v posamezni študiji se je uporabilo barvno kodiranje. Naj na tem mestu ponovno povzamemo barvno shemo: zelena predstavlja pojme, ki so bili temeljito obravnavani v študiji, rumena predstavlja pojme, ki so bili le omenjeni, rdeča pa pojme, ki jih študija ni obravnavala.

Tudi kodirna shema se je ohranila enaka, kot v [6], saj smo s tem želeli zagotoviti primerljivost med študijama.

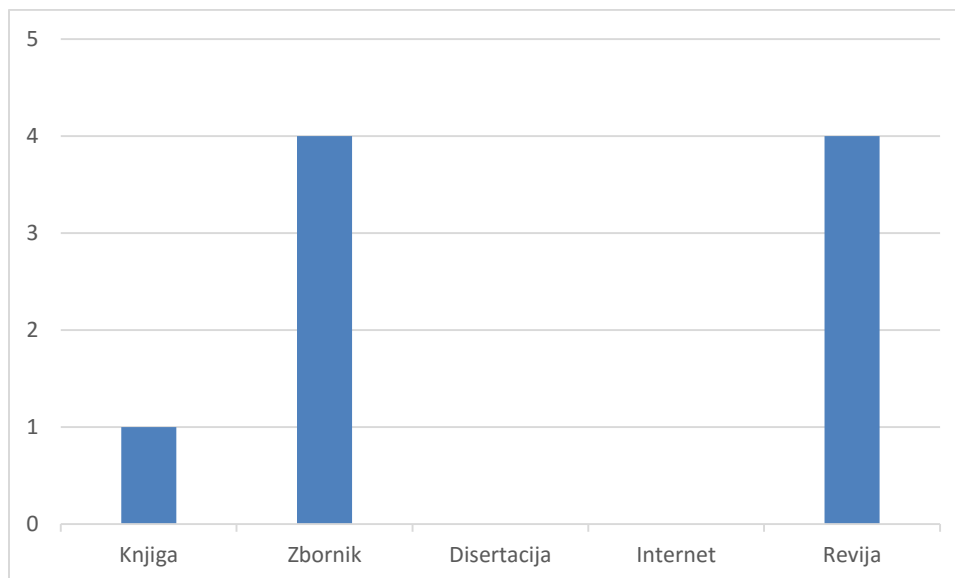
4.2.2 Obrazec za pridobivanje podatkov

Obrazec je nastal na osnovi podatkov, ki so bili predstavljeni v rezultatih raziskave [6]. Deli obrazca, ki pokrivajo tematiko posameznega raziskovalnega vprašanja so na obrazcu označeni. Obrazec je predstavljen v prilogi Priloga VI – Predloga obrazca za pridobivanje podatkov.

4.3 Predstavitev rezultatov

4.3.1 Rezultati iskanja in ocenjevanja kakovosti literature

Ker smo se v naši raziskavi omejili zgolj na znanstveno literaturo, je bilo rezultatov iskanja bistveno manj kot v študiji [6]. Skupno smo odkrili 43 primarnih študij, od tega jih je bilo 41 pridobljeno z iskanjem po elektronskih bazah podatkov, dve pa z ročnim pregledom revije "Empirical Software Engineering". Po postopku, ki je prikazan na sliki Slika 8, smo preverili skladnost s kriteriji za vključitev študije v raziskavo. Odkrili smo, da 5 študij ni ustrezalo kriteriju "*V študiji mora biti jasno razvidno, da se osredotoča na razvoj programske opreme z uporabo Kanban pristopa ali vitkega razvoja.*", 11 študij ni ustrezalo kriteriju "*Študija naj opisuje elemente in pristope, ki se uporabljajo za implementacijo Kanban*", in 18 študij ni ustrezalo kriteriju "*Če je bila študija objavljena večkrat, v več različnih virih, se v obravnavo vključi le nazadnje objavljena študija*". Za nadaljnjo obravnavo je tako ostalo 9 študij, med njimi je tudi članek [6], ki je bil objavljen v letu 2015. Na sliki Slika 19 je prikazana porazdelitev študij po vrsti gradiva. Internetnih virov, ki spadajo med sivo literaturo, pričakovano ni na grafu, rubriko pa smo vseeno ohranili zaradi konsistentnosti s [6].

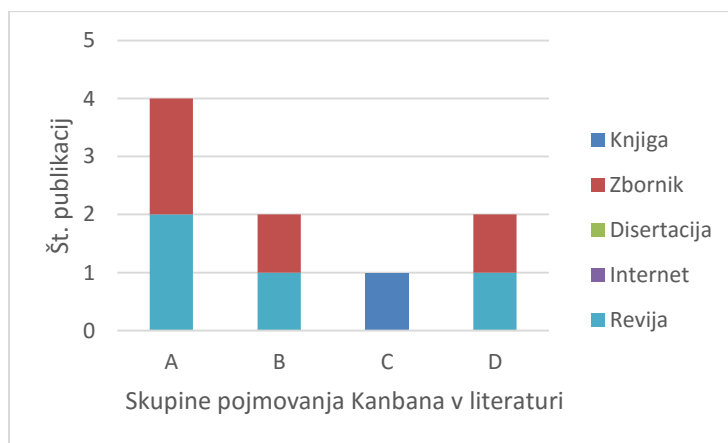


Slika 19: Porazdelitev primarnih študij glede na vrsto gradiva

4.3.2 Rezultati raziskave

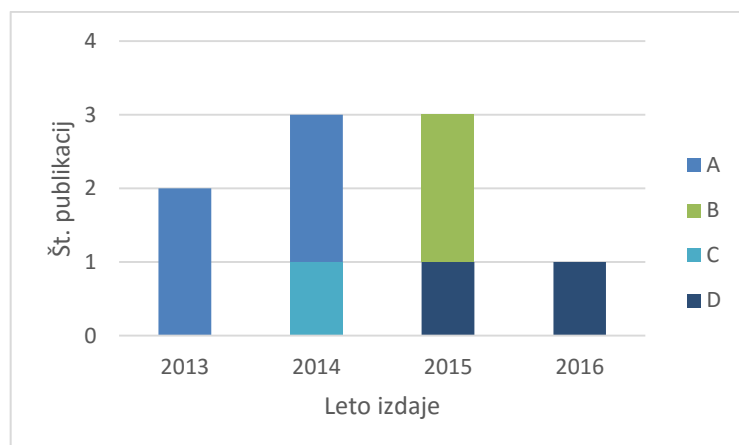
Primarne študije smo kategorizirali v štiri skupine, ki se ločijo po načinu obravnave Kanbana in so navedene v tabeli Tabela 3. Na sliki Slika 20 smo prikazali graf porazdelitve primarnih študij po skupinah po tipu študije.

Čeprav je vzorec majhen, se lahko iz grafa razbere, da se trenutno strokovnjaki in praktiki ukvarjajo z doseganjem vitkosti v organizaciji z uporabo Kanbana kot orodja.



Slika 20: Porazdelitev primarnih študij po skupinah pojmovanja Kanban in po tipu študije

Porazdelitev primarnih študij po skupinah pojmovanja Kanbana in letu objave je prikazano na sliki Slika 21.



Slika 21: Prikaz primarnih študij po skupinah pojmovanja Kanbana in letu objave.

Graf sicer nakazuje na porast števila publikacij z leti, vendar je treba tudi v tem primeru upoštevati velikost, oz. majhnost vzorca. Iskanje primarnih študij je bilo izvedeno v mesecu maju 2016, zato števila najdenih publikacij v tem letu ne moremo primerjati z ostalimi leti.

4.3.3 Elementi Kanbana

V nadaljevanju so opisane ugotovitve o načinu obravnave elementov Kanbana v primarnih študijah. Elementi Kanbana so bili prevzeti iz [6], podrobno so opisani v poglavju 3.3.6.3, zato njihovih definicij na tem mestu ne bomo ponavljali, ampak bomo zapisali le ugotovitve. Med primarnimi študijami smo obdelali tudi članek [6] z oznako [SI2], ki je podrobno predstavljen v poglavju Poglavlje 3. Ugotovitev iz članka na tem mestu ne bomo ponovno obravnavali.

Na slikah Slika 22, Slika 23, Slika 24 in Slika 25 so prikazani podatki o obravnavi elementov Kanbana za vsako skupino primarnih študij posebej. Naj ponovno predstavimo barvno shemo: zelena predstavlja pojme, ki so bili temeljito obravnavani v študiji, rumena predstavlja pojme, ki so bili le omenjeni, rdeča pa pojme, ki jih študija ni obravnavala.

V nobeni izmed primarnih študij, razen v [SI2] niso bili podrobneje obravnavani elementi: C5 Kriteriji vključitve, C7 Zaključena delovna nalog, C9 Validirano učenje, C12 Ozko grlo, C13 Čakalna vrsta in C17 Avatar, zato se spodaj ne navajajo.

C1 Metoda Kanban. V [SI1] in [SI8] so avtorji podali podrobno definicijo Kanban metode, ki jo povzemajo po [2]. V [SI1] so nadalje navedli cilj implementacije Kanban metode, ki je nudenje pregleda nad delovnim procesom (angl. provide visibility) in izboljšanje sporazumevanja ter sodelovanja na nivoju celotnega razvojnega in vzdrževalnega procesa. V

primarni študiji [SI5] se navaja, da je poudarek Kanban metode na razporejanju dela (angl. scheduling of work) tako, da je možna dostava izdelkov ravno ob pravem času, JIT. Za lastnosti, ki Kanban metodologijo ločijo od agilnih so navedene: Kanban tabla, maksimiranje produktivnosti, zvezna dostava (angl. continuous delivery), minimiziranje odpadkov in omejitev obsega dela v teku WIP. V [SI9] avtorji ugotavljajo, da se Kanban čedalje več uporablja za vizualizacijo delovnega toka, merjenje potrebnega časa in določanje omejitev obsega dela v teku. Pri implementaciji Kanban metode so avtorji obravnavali možnosti vizualizacije vrst (angl. visualize queue), tehnike za iskanje primarnega vzroka problema, ter spodbujali širjenje znanja in učenje znotraj organizacije. Kanban se v organizaciji, v kateri so bili pridobljeni empirični podatki, čedalje več uporablja za vizualizacijo in upravljanje z WIP, ter za vzpostavljanje izvlečnega delovanja (angl. create a pull culture).

C2 Kanban tabla. Avtorji v [SI1] navajajo, da se načela Kanban metode uresničujejo s Kanban tablo, ki vizualizira tok aktivnosti procesa v stolpcih. Za prikazovanje delovnih nalog in s tem trenutnega stanja se uporabljajo kartice. Tok delovnih nalog skozi proces je optimiziran z uporabo WIP omejitev na vsakem stolpcu z aktivnostjo tako, da je omogočen največji pretok nalog, ki so povlečene skozi sistem. Kanban tabla primerjajo s Scrum tablo in ugotavljajo tri bistvene razlike: 1) Scrum tabla ima dva načina delovanja – planiranje in izvedba, Kanban tabla nima posebnega načina delovanja za planiranje; 2) obe tabli uporabljata podobno zasnovo z aktivnostmi v stolpcih, le da se na Kanban tabli delo ne razporeja po iteracijah – šprintih (angl. Scrum sprint), Kanban tabla prikazuje delovne naloge, ki se lahko povlečejo v izvajanje kadarkoli, seveda skladno z razpoložljivimi resursi (upoštevanje omejitev WIP); 3) Scrum tabla se po vsaki iteraciji ponastavi, medtem ko se Kanban tabla načeloma ne ponastavlja nikoli. V [SI6] so ob uvajanju Kanbana za začetno zasnovo table prevzeli korake procesa, ki so ga do tedaj upravljali s Scrumom. Nato so aktivnostim določili omejitve WIP ter tablo v več iteracijah dopolnjevali.

C3 Izvlečni sistem. V [SI6] avtorji trdijo, da omejitve dela v teku implicirajo izvlečni sistem, ki deluje kot eden izmed glavnih spodbujevalcev zveznih, inkrementalnih in evolucijskih sprememb sistema. V študiji [SI7], ki se osredotoča na stanje uporabe vitkega razvoja na področju razvoja informacijskih sistemov velikih razsežnosti, ugotavljajo, da kljub temu, da je izvlečni sistem kot element Kanbana eno izmed najpogostejše obravnavanih orodij vitkosti, njegove uporabe na področju razvoja informacijskih sistemov velikih razsežnosti še ni zaznati. V [SI8] so avtorji obravnavali izdelavo na zahtevo (angl. pull). Sistem z izdelavo na zahtevo določa, da je vse delo narejeno takrat, ko nastane zahteva zanj. Zahteva kupca, oz. trg postane tako glavno gonilo dela, nepotrebnosti in nedokončani izdelki pa morajo biti odpravljeni. Kot predpogoje za implementacijo toka in izdelave na zahtevo navajajo: 1) v organizaciji ne sme obstajati pojav funkcionalnega silosa, ki pomeni odtujenost posameznikov oz. skupin od

celote, 2) kratki cikli s povratno informacijo in ustrezno pokritostjo testiranj zvezne integracije, 3) omejitve obsega dela v teku na vseh nivojih organizacije, od vodstva, do razvijalcev, ki naj omogoča ohranjanje znosnega tempa in 4) zavedanje o medsebojni sodvisnosti funkcionalnosti.

C4 Razvrščanje po prioriteti. Tematika primarne študije [SI1] je prehod od uporabe Scruma k uporabi Kanbana ekip, ki se ukvarjajo z vzdrževanjem programske opreme. Avtorji študije ugotavljajo, da se pri vzdrževalnem delu uporabniške zahteve pojavljajo dnevno in nedeterministično. Prioriteta delovnih nalog se stalno spreminja skladno z nivoji kritičnosti zahtev. Po uvedbi Kanbana in Kanban table je shajanje s takim pogostim spreminjanjem prioritet delovnih nalog bistveno olajšano. Kanban tabla nudi pregled nad vsemi uporabniškimi zahtevami, kar je nujno za učinkovito določanje prioritet nalogam. Ekipa, ki uporablja Kanban, začne delo vedno na delovni nalogi z najvišjo prioriteto. Ko je naloga zaključena, je takoj dostavljena kupcu. S stalnim popravljanjem prioritet nalog in z zvezno dostavo postanejo ekipe, ki opravljajo naloge vzdrževanja, izredno odzivne. Kot učinkovito tehniko za določanje prioritet navajajo avtorji v [SI3] tehniko MoSCoW – Must do it, Should do it, Could do it, Won't do it. Tehnika je podrobneje razložena v [19]. V [SI7] avtorji navajajo, da so bile prioritete delovnih nalog določene v sodelovanju s kupcem. Pri določanju prioritet so uporabljali model, ki ga je Kano opisal v [20] in opisuje razmerje med zadovoljstvom kupca in kakovostjo, ter kategoriziranje kupčevih zahtev v tri kategorije.

ID	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14	C15	C16	C17	C18	C19	C20
SI3	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
SI7	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
SI8	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
SI9	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●

Slika 22: Obravnava elementov Kanbana v primarnih študijah tipa A.

C6 Omejitev obsega dela. Analiza primarnih študij je pokazala, da je omejitev obsega dela pomemben element Kanbana. Obravnavan je v vseh primarnih študijah, razen v eni. Večina primarnih študij ugotavlja, da kot element Kanbana igra ključno vlogo pri doseganju zveznega toka (angl. continuous flow) in uvedbi izvlečnega sistema (angl. pull system). V [SI1] so avtorji ugotovili, da so bili razvijalci v primeru (pre)nizke vrednosti omejitve WIP pogosto v stanju brezdelja (angl. idle), v primeru previsoke vrednosti pa je prihajalo do prepogostega prehajanja med delovnimi nalogami (angl. task switching), kar je imelo za posledico daljšanje časa med dostavami izdelkov. Z izkušnjami so člani ekip ugotovili omejitve WIP, ki so jim najbolj ustrezale, in se jih striktno držali. Z zmanjšanjem WIP omejitev so uspeli zmanjšati čakalne vrste (angl. queues) med aktivnostmi. V [SI8] so avtorji navedli, da kljub temu, da so

ekipe v organizaciji po kompetencah na približno enakem nivoju, prihaja do razlik v načinu izvajanja dela (angl. teamwork) in zmožnosti (angl. capacity). Zaradi tega je morala vsaka ekipa s poskušanjem ugotoviti omejitve WIP, ki so njenim članom najbolj ustrezale.

C8 Vzvratno pomikajoča se delovna naloga. Element je bil obravnavan le v študiji [SI3]. Element je obravnavan s primerom, ko tester vrne delovno nalogo v izvedbo razvijalcu, v primeru, da so na nalogi potrebni popravki. Tester nalogi določi prioriteto in jo postavi na seznam zahtev na Kanban tabli. Nato sta možna dva scenarija: 1) ko je naloga po prioriteti na najvišjem mestu in omejitev WIP še ni dosežena, se lahko začne delo na nalogi; 2) razvijalec odloži nalogo, ki jo izvaja, v čakalno vrsto (angl. queue), ter povleče nalogo, na kateri je tester odkril nepravilnosti. Tudi v tem primeru je treba upoštevati prioritete nalog in omejitve WIP.

C10 Čas cikla / potrebni čas. Primarne študije navajajo čas cikla in potrebni čas kot osnovne metrike Kanbana, ki se uporabljajo pri planiranju in izboljševanju procesov. V študiji [SI3] se element navaja kot ena izmed metrik pri upravljanju z odpadki v procesu. Kot primer odpadka navajajo centralizirano sprejemanje odločitev, ki podaljšuje potrebni čas. Podrobneje je čas cikla obravnavan v študiji [SI4]. Študija navaja smernice za merjenje časa cikla, ki naj poteka v štirih korakih: 1) zajem datuma nastanka (ideje) zahteve (angl. created date); 2) zajem datuma začetka izvajanja dela na zahtevi – to je takrat, ko se sprejme odločitev, da bo zahteva res implementirana, ter se ji določi prioriteta (angl. started date); 3) zajem datuma zaključka dela na zahtevi – zahteva, oz. sedaj funkcionalnost je pripravljena na dostavo kupcu (angl. completed date); in 4) zajem datuma, ko je funkcionalnost na voljo kupcu za uporabo (angl. in production date). Čas cikla je opredeljen kot razlika: *completed date* – *started date* in se v študiji določa kot glavni kazalnik uspešnosti uvedbe nove metodologije v delovni proces. Avtorji so v članku dokazali, da se za napovedovanje časa cikla uporablja Weibullova porazdelitev in podajajo naslednje ugotovitve: 1) izračun časa cikla naj ne temelji na normalni porazdelitvi, 2) uvedba agilnih metodologij je porazdelitvi časa cikla spremenila velikostni in oblikovni parameter (angl. scale and shape), tako da se je zmanjšala variabilnost (angl. variability), in 3) velikostni in oblikovni parameter časa cikla sta temelj za podajanje utemeljenih smernic za razvoj procesa (angl. informed process recommendations). Z verjetnostnim modeliranjem je omogočeno ocenjevanje finančnega aspekta uvedbe nove razvojne metodologije. V študiji se omenja tudi potrebni čas, ki je definiran kot razlika: *in production date* – *created date*, a se mu posebna pozornost ne posveča.

ID	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14	C15	C16	C17	C18	C19	C20
SI4	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
SI5	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●

Slika 23: Obravnava elementov Kanbana v primarnih študijah tipa B.

C11 Orodje za merjenje performas. V primarnih študijah se kot najpogostejši metriki za merjenje performans, navajata čas cikla in potrebni čas. Vzrok za pogosto uporabo je preprosto beleženje in spremljanje. Kot orodji za merjenje performans se v [SI6] navajata: 1) dejansko porabljeni čas (angl. Perfect Engineering Hours, PEH) je čas, ki bi ga razvijalec porabil za izvedbo delovne naloge, če bi bil pri tem osredotočen le na izvajanje naloge, drugih motečih elementov iz okolice pa ne bi bilo. Gre za subjektivno mero kompleksnosti uporabniške zgodbe. 2) diagram preostalega dela (angl. burn down charts) predstavlja produktivnost v odvisnosti od časa, oz. spreminjanje števila nedokončanih delovnih nalog s časom. Število delovnih nalog, ki še niso dokončane, se prikazuje na ordinatni osi, na abscisi pa čas. Diagram je v pomoč pri predvidevanju, kdaj bo delo zaključeno. Poleg tega se diagram lahko uporablja za prikaz začetno planiranega, ponovno planiranega (planiranje se izvede ponovno, npr. po zaključeni iteraciji) in dejansko izvedenega dela. V primarni študiji [SI7] sta predstavljeni dve metriki: prva služi ocenjevanju števila napak (angl. defect) na projektih razvoja programske opreme velikih razsežnosti, ki temelji na različnih elementih vitkosti in agilnosti [21]. Druga metrika služi odkrivanju neželenih pojavov, kot so ozka grla, s spremljanjem števila zahtev v inventarju (angl. inventory), ki so bile prikazane na kumulativnem diagramu poteka [22].

C14 Odkrivanje odpadkov. Avtorji primarnih študij ugotavljajo, da je odkrivanje odpadkov eden izmed pomembnejših elementov Kanbana in vitkosti. Najbolj podrobno je bil element (sicer v kontekstu vitkosti) obravnavan v primarni študiji [SI3]. Avtorji v študiji ugotavljajo, da "klasični" sezname vzrokov, oz. virov odpadkov, ki so bili prevzeti s področja proizvodne industrije, ne ustrezajo za področje razvoja programske opreme. V študiji so poskušali na podlagi empiričnih raziskav identificirati najpogostejše vzroke za odpadke, ki se pojavljajo v procesu razvoja programske opreme in tako predstavili predlog klasifikacije odpadkov vitkega razvoja programske opreme z devetimi klasifikacijskimi skupinami, ki so nadalje razdelane v 42 posameznih virov odpadkov. Za vsakega od vzrokov za nastanek odpadkov so predlagali tudi ukrep, ki odpadke odpravlja. V primarni študiji [SI7] so avtorji ugotavljali navedbe v literaturi o uporabi 13 načel vitkosti po Morganu in Likerju [23] v praksi. Odkrili so sedem vrst odpadkov, ki jih obravnavajo primarne študije v njihovi raziskavi: 1) prevelika proizvodnja (angl. overproduction); 2) čakanje (angl. waiting) – na material, informacije, sprejem odločitev; 3) transport (angl. conveyance) – nepotrebno pomikanje delovnih predmetov; 4) odvečno delo (angl. processing) – izvajanje odvečnega dela na delovni nalogi, ali izvajanje nepotrebnih delovnih nalog; 5) zaloga (angl. inventory) – kopičenje materiala ali informacij, ki niso potrebne; 6) premiki (angl. motion) – odvečni premiki ali aktivnosti med izvajanjem delovnih nalog; 7) popravki (angl. correction) – odpravljanje težav. V [SI8] so odpadke definirali kot vsako porabljanje resursov, ki ne proizvede vrednosti za kupca. Vitki razvoj skrbi za nenehno odkrivanje in odpravljanje odpadkov. V raziskavi so odkrili, da se v

vitkem razvoju glavni poudarek daje načelu vrednosti za kupca (angl. value) in vgradnji kakovosti (angl. build quality), v nasprotju s splošnim prepričanjem, da je bistvo vitkega razvoja izločanje odpadkov. V [SI9] se kot najpogostejše vrste odpadkov, oz. vzrokov za nastanek odpadkov navajajo: prepozno sprejete odločitve, previsoke vrednosti omejitev WIP in nepotrebne funkcionalnosti.

C15 Sodelovanje v ekipi. Primarne študije, v katerih je bil element obravnavan, navajajo po vpeljavi Kanbana splošno izboljšanje v sodelovanju tako znotraj ekipe, kot tudi med ekipami. V [SI1] avtorji posebej navajajo napredek v dojemanju lastnih zadolžitev in zadolžitev znotraj ekipe in tudi navzven, na medekipnem nivoju. V [SI8] so na podlagi empirične raziskave ugotovili, da se, kljub povečani transparentnosti v procesu razvoja po vpeljavi vitkosti, še vedno pojavlja tendenca k omejenemu sodelovanju med ekipami, ki pokriva le najnujnejše potrebe za delovanja (sicer v preteklosti uporabljene) agilne metodologije. V obravnavani organizaciji so ob vpeljavi vitkosti poskušali izboljšati sodelovanje in okrepiti občutek pripadnosti ekipi s preoblikovanjem delovnih prostorov – posamezne pisarne so zamenjali nadomestili z večjimi prostori, primernimi za delo ekip, delovna mesta razvijalcev so postavili za skupne mize, uvedli so uporabo raznih pripomočkov, kot so table, flipchart table, zasloni in wiki, ki so namenjeni porazdeljenim ekipam. V [SI9] avtorji navajajo, da je v organizaciji, ki je bila obravnavana v empirični raziskavi, ekipno delo izrednega pomena. Za doseganje večje povezanosti posameznikov znotraj ekipe in sledenja skupnih ciljev so v organizaciji izkoristili naravo delovanja metod Scrum in Kanban.

ID	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14	C15	C16	C17	C18	C19	C20
SI6	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●

Slika 24: Obravnava elementov Kanbana v primarni študiji tipa C.

C16 Sestanki in njihova struktura so bili obdelani v petih primarnih študijah: [SI1], [SI3], [SI6], [SI7] in [SI9]. V študijah se je v vseh primerih omenjalo uporabo kratkih dnevnih sestankov (angl. daily stand-up meetings) in sestanke v okviru aktivnosti retrospektive. Dnevni sestanki se uporabljajo za razglabljanje o delu, ki je trenutno v teku, za kratkoročno planiranje in obravnavo (spremembe) prioritet delovnih nalog. V [SI1] so se ob vpeljavi Kanbana začeli posluževati tako kratkih dnevnih sestankov v okviru posamezne ekipe, kot tudi sestankov z drugimi ekipami razvijalcev in predstavniki vodstva, ki so se odvijali dvakrat tedensko. V [SI3] so avtorji navedli, da so v obravnavani organizaciji omejili odpadke, ki izvirajo iz sestankovanja tako, da so za izvajanje sestankov določili pravila: 1) določiti je treba cilje sestanka; 2) objavljene je treba o sestanku obvestiti vsaj dva delovna dneva pred sestankom, ter v vabilu navesti agendo sestanka, 3) določiti je treba omejitve razprav, 4) teme

izven obsega agende sestanka se zabeleži za obravnave na sledečih sestankih, 5) zapisati je treba jasne zabeležke sestankov, 6) razpoznati je treba opravila s specifičnimi roki in načrti izvedbe, in 7) izvajanje pregleda ter ocenjevanja uspešnosti sestankov.

C18 Planiranje in ocenjevanje. Ugotovitve v primarnih študijah o planiranju in ocenjevanju so skope. V [SI3] se omenja pomembnost pomembnost ustreznega ocenjevanja obsega projekta, ki naj vključuje tudi sredstva, ki bodo omogočala izvedbo testiranja. Kot navajajo, se natančnost ocenjevanja obsega povečuje z izkušnjami ocenjevalcev. Pri ocenjevanju obsega projekta svetujejo sodelovalno sprejemanje odločitev (angl. collaborative decision-making) in uporabo informacij, ki so nastale med izvajanjem podobnih projektov (angl. historical data). Če takšne informacije niso na voljo, avtorji članka svetujejo, da se ubere ocenjevalna strategija od spodaj navzgor (angl. bottom-up estimation strategy). Pomembnost planiranja je izpostavljena v študiji [SI9], v kateri avtorji svetujejo planiranje sprinov, (angl. sprint planning), planiranje izdaje (angl. release planning), v primeru večjih projektov pa tudi vnaprejšnje planiranje (upfront planning). Pomembnost planiranja se navaja predvsem v kontekstu izdelave zasnove in arhitekture sistema.

C19 Politika Kanban pristopa. V primarnih študijah se ta element obravnava v majhnem obsegu in navadno v širšem pomenu, torej politike ne le v kontekstu Kanban pristopa. V študiji [SI1] so avtorji predstavili politike za obvladovanje pogostega spreminjanja prioritet delovnih nalog z uporabo Kanban pristopa: 1) ne začenjaj opravljati dela, če zanj ne obstaja prikaz delovne naloge na Kanban tabli; 2) o poteku dela na delovni nalogi in o prioritetah razpravljaj na dnevnih sestankih; 3) iz Kanban table naj se povlečejo delovne naloge z upoštevanjem najvišje prioritete; 4) ko se iz Kanban table povleče nalogo z visoko prioriteto, naj se naslednja naloga povleče šele, ko je prva zaključena; 5) v primeru pojava nujne zahteve (angl. expedite) naj se ta postavi na "hitri pas" (angl. express lane) Kanban table, delo na njej pa naj opravlja celotna ekipa dokler naloga ni zaključena. V študiji se Kanban metoda navaja kot idealna za okolje, v katerem se prioritete nalog hitro spreminjajo. V študiji [SI7] je obravnavana organizacijska politika in standardi, kot osnova oz. ogrodje procesa razvoja programske opreme.

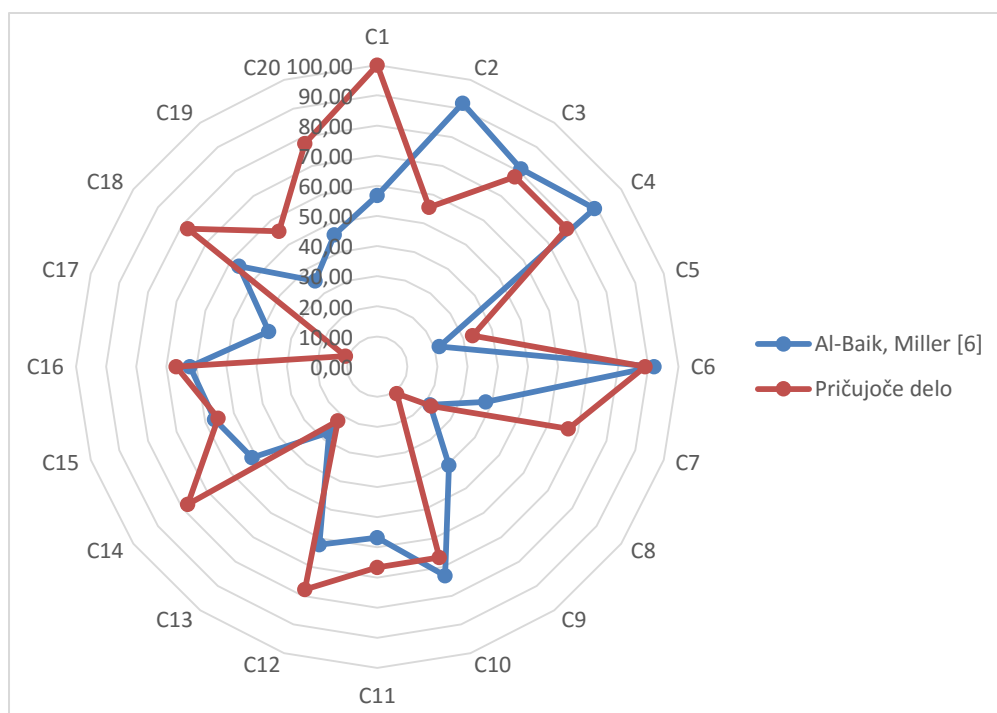
C20 Povratna informacija. V primarni študiji [SI3] so kot enega izmed vzrokov za nastanek odpadkov obravnavali majhno vpletenost kupca v proces razvoja programske opreme. Kot najpogostejši razlog za majhno vpletenost kupca so avtorji navedli odsotnost nekega uveljavljenega načina pristopa do kupca. V izogib temu so avtorji predlagali dve politiki za pridobivanje povratne informacije, ki se razlikujeta gleda na nujnost oz. obseg potrebne povratne informacije. Po prvi politiki se kupca oz. njegove predstavnike vključi v proces razvoja na dnevnem nivoju z uporabo principov metodologije ekstremnega programiranja

(angl. eXtreme Programming), pri kateri kupec praktično postane član razvojne ekipe. Po drugi politiki, ki se uporablja, ko se kaže potreba po razmeroma majnem obsegu povratne informacije, pa se kupca vpleta le občasno, pri čemer se povratne informacije pridobivajo z razvojem prototipov. V [SI6] so izpostavili pomen povratne informacije o uporabnosti izdelka, ki je botrovala izboljšanju procesa razvoja programske opreme. V [SI7] so avtorji poudarili pomen razširjanja informacije o tem, kaj za kupca predstavlja dodano vrednost, po celotni organizaciji, s čimer se zmanjšuje verjetnost za nastanek odpadkov. V [SI9] so avtorji na obravnavanem primeru iz prakse ugotavljali, da se po vpeljavi Kanbana na kratke cikle pridobivanja povratne informacije (angl. feedback loops) še vedno gleda kot na izziv.

ID	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14	C15	C16	C17	C18	C19	C20
SI1	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
SI2	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●

Slika 25: Obravnava elementov Kanbana v primarnih študijah tipa D.

Na spodnjem grafu je prikazana pogostost obravnave posameznega elementa v odstotkih v primarnih študijah v sistematičnem pregledu, ki sta ga opravila Al-Baik in Miller [6], ter v pričujočem delu. Iz grafa lahko razberemo rahel porast obravnave elementov Kanbana v obravnavani literaturi.



Slika 26: Primerjava pogostosti obravnave elementov Kanbana v literaturi glede na sistematični pregled

4.3.4 Koristi in izzivi uporabe Kanban pristopa

V analizi primarnih študij smo ugotovili, da se nekatere koristi in izzivi, ki jih v organizacijo prinaša Kanban pristop in so bile identificirane v literaturi v obdobju pred 2012 [6], v primarnih študijah še vedno navajajo. V nadaljevanju sta v tabelah Tabela 6 in Tabela 7 prikazana seznama koristi in izzivov uporabe Kanban pristopa.

Koristi vpeljave Kanban pristopa	Obravnava v primarni študiji	Odstotek primarnih študij v %
Izboljšuje vizualni nadzor in s tem podpira in olajša sprejemanje odločitev	[SI1] [SI2] [SI3] [SI9]	44,4
Olajša usklajevanje večfunkcijskega ekipnega dela in uvaja samoorganizacijo	[SI1] [SI2] [SI8] [SI9]	44,4
Spodbuja vpeljavo ciklov kakovosti in opravil, ki zagotavljajo kaizen (postopek nenehnega izboljševanja)	[SI9] [SI2]	22,2
Zmanjšuje čas cikla, oz. potrebni čas	[SI1] [SI2] [SI3] [SI9]	44,4
Povečuje zadovoljstvo kupca in omogoča ustvarjanje izdelkov visokega cenovnega razreda	[SI1] [SI2] [SI5] [SI3] [SI7]	55,6
Zmanjšuje tehnična tveganja in tveganja, ki jih prinaša tržišče	[SI2] [SI3]	22,2
Spodbuja razvoj strategij nenehnega izboljševanja	[SI2] [SI6] [SI8]	33,3
Povečuje predvidljivost izdaje novih izdelkov kupcu z upoštevanjem spreminjanja kupčevih zahtev	[SI1] [SI2] [SI5] [SI7] [SI9]	55,6
Zagotavlja razvoj znanja in spodbuja povezanost članov ekip	[SI1] [SI2] [SI3] [SI8] [SI9]	55,6

Usmerja in olajšuje upravljanje z organizacijskimi spremembami in spremembami organizacijske kulture	[SI2] [SI3]	22,2
Povečuje kakovost izdelkov, kar se kaže v zmanjšanju števila okvar, povečanju števila izdelkov, ki ustrezajo kakovostnim merilom, in zmanjšanju števila napak.	[SI1] [SI2] [SI3]	33,3

Tabela 6: Koristi, ugotovljene v članku [6], ki so bile obravnavane v primarnih študijah.

Izzivi pri vpeljavi Kanban pristopa	Obravnava v primarni študiji	Odstotek primarnih študij v %
Težko je določiti ustrezne metrike. Na primer potrebni čas (angl. lead time) se je za merjenje performans izkazal kot težaven	[SI2]	11,1
Vpeljava Kanban pristopa zahteva temeljito poznavanje konceptov, načel in praks vitkosti	[SI2]	11,1
Uporabljajo se nejasne definicije osnovnih elementov Kanban, kot na primer definicija zaključene delovne naloge	[SI2]	11,1
Ni ustreznih smernic za vpeljavo pristopa in elementov Kanban.	[SI1] [SI2] [SI5] [SI9]	44,4
IT organizacije s težavo vpeljujejo nekatere elemente Kanban pristopa, kot je npr. vzpostavljanje zveznega toka. (angl. flow)	[SI2] [SI3] [SI8]	33,3
Treba je spremeniti organizacijo in strukturo ekip razvijalcev	[SI2]	11,1
Vpeljava Kanban pristopa v proces razvoja programske opreme in njegova integracija z obstoječimi agilnimi tehnikami je zapletena, draga in časovno potratna	[SI2]	11,1

Tabela 7: Izzivi, ugotovljeni v članku [6], ki so bile obravnavane v primarnih študijah.

Koristi vpeljave Kanban pristopa	Obravnava v primarni študiji	Odstotek primarnih študij v %
Zmanjšuje stroške in povečuje propustnost in učinkovitost procesa razvoja programske opreme	[SI1] [SI5] [SI6] [SI8] [SI9]	55,6
Olajša razporejanje razvijalcev med ekipami	[SI1]	11,1
Spodbuja osredotočenost k opravljanju zadolženih delovnih nalog	[SI1] [SI3]	22,2
Spodbuja sodelovanje v razvojnem procesu s strani kupca, ki podajanja koristne povratne informacije, te pa so učinkovito razpršene vsem zaposlenim v organizaciji, ki jih potrebujejo	[SI5] [SI6] [SI8] [SI9]	44,4
Povečuje motiviranost zaposlenih pri delu in spodbuja ekipno delo	[SI8]	11,1

Tabela 8: Koristi vpeljave Kanbana, ki niso bile obravnavane v članku [6].

Izzivi pri vpeljavi Kanban pristopa	Obravnava v primarni študiji	Odstotek primarnih študij v %
Proces sprejemanja odločitev otežuje doseganje konstantega toka	[SI8]	11,1
Določitev in uporaba standardov v procesu razvoja programske opreme je težavna zaradi raznolikosti delovnih nalog	[SI8]	11,1
Za doseganje konstantega toka mora implementacija Kanbana poleg delovnega procesa razvojnega oddelka vključevati tudi druge oddelke organizacije	[SI8]	11,1
Sprememba miselnosti razvijalcev iz delovanja v načinu <i>push</i> v <i>pull</i> se je izkazala kot zahtevna	[SI8]	11,1

Tabela 9: Izzivi vpeljave Kanbana, ki niso bile obravnavane v članku [6].

Iz zgornjih tabel lahko razberemo, da so se avtorji primarnih študij bolj osredotočili na poročanje koristi, kot pa izzivov. Večina koristi in izzivov, s katerimi sta se srečala avtorja v sistematičnem pregledu literature v [6] se pojavlja tudi v primarnih študijah, ki so bile obravnavane v tem sistematičnem pregledu. Poleg teh smo v tabelah Tabela 8 in Tabela 9 našli še nekaj novo odkritih.

4.3.5 Jasnost obstoječih smernic za vpeljavo Kanban pristopa v organizacijo

Kanban se obravnava v vseh primarnih študijah. Zanimivo je, da se vedno uporablja z veliko začetnico, torej kot Kanban metoda. Uporabi besede z malo začetnico, ki pomeni vrsto izvlečnega sistema, so se avtorji izognili. Avtorji v osmih primarnih študijah ([SI1], [SI2], [SI3], [SI4], [SI5], [SI6], [SI8] in [SI9]) razlikujejo Kanban od kanban sistema. V študiji [SI7] pa sta pojma pomešana. Kanban se navaja kot izvlečni sistem: "*... največkrat omenjena praksa (angl. practice) vitkega razvoja je izvlečni sistem (Kanban razvoj programske opreme)...*", in kot metodo: "*...vitki razvoj vključuje metode, kot so Kanban, Kaizen (nenehno izboljševanje), hkratni razvoj (angl. concurrent development), itd. ...*".

V študijah [SI4] in [SI5] se Kanban ne uporablja za pojmovanje Kanban metode, kot jo definira Anderson v [2], ampak za pojmovanje metodologije, in to agilne. V študiji [SI6] se sicer beseda pravilno uporablja za označevanje Kanban metode, vendar je nekajkrat uporabljena tudi za pojmovanje Kanban table. Uporaba izraza Kanban za pojmovanje Kanban table lahko kaže na nekonsistentnost uporabe ali celo napačno razumevanje izraza. Po drugi strani gre lahko zgolj za površnost.

V članku [6] so avtorji predlagali, da se vpeljave Kanbana v organizacijo lotimo v treh korakih: 1) vizualizacija delovnega toka; 2) določitev omejitev obsega dela v teku in 3) Merjenje in upravljanje delovnega toka. V nobeni primarni študiji postopek implementacije Kanbana ni bil jasno opredeljen ali raziskan.

V [SI1] so avtorji identificirali slabosti obstoječe metodologije razvoja v uporabi v organizaciji (Scrum) in nato za vsak izziv podali kako ga implementacija Kanbana odpravlja. Smernice za implementacijo kot take niso podane.

V članku [SI3] so se avtorji osredotočili na odpravljanje odpadkov v procesu razvoja programske opreme. Postopek odkrivanja in odpravljanja odpadkov je dobro določen.

Študija [SI4] obravnava merjenje in napovedovanje učinkov vpeljave nove metodologije. Za metriko se uporablja čas cikla. Samemu postopku vpeljave metodologije avtorji niso posvetili pozornosti, zanimalo jih je stanje pred in po vpeljavi metodologije v organizaciji.

Študija [SI6] zgoraj omenjene korake vpeljave sicer omenja, niso pa naštet v zaporedju. Za uresničevanje prvega koraka navajajo uporabo Kanban table. Za metrike, ki naj se uporabljajo v tretjem koraku pa navajajo: 1) metrike, ki prikazujejo implementacijo tacitnega znanja (število dnevnih sestankov, število sestankov z naročnikom, število retrospektivnih aktivnosti, število izboljšanih ali dodanih procesov) in 2) metrike produktivnosti razojnih ekip - dejanski porabljeni čas (angl. Perfect Engineering Hours) in diagrami preostalega dela.

V [SI7] navajajo, da se implementacija vitkega razvoja običajno začne z vizualizacijo.

V študiji [SI8] so podrobno opisana orodja, ki so bila implementirana za doseganje vitkosti v organizaciji. Koraki implementacije pa niso kronološko urejeni, ampak se podajajo le na način kaj je bilo implementirano in s katerim namenom.

Tudi v študiji [SI9] so zgoraj omenjeni koraki implementacije Kanbana obravnavani, vendar le kot elementi Kanbana, ne pa kot smernice za njegovo implementacijo.

Članek [SI5] korakov implementacije sploh ne omenja.

V članku [6] sta avtorja podala naslednje tri ovire implementacije Kanbana, ki sta jih odkrila v svojem sistematičnem pregledu literature:

- a) Neskladje definicij: za elemente ne obstaja standardna definicija v razpoložljivi literaturi,
- b) Nejasne smernice: slabo določena navodila in smernice za oblikovanje in implementacijo elementov, in
- c) Protislovja in neujemanja: elementi so v protislovju, oz. se njihova definicija ne ujema z drugim elementom ali načelom vitkosti.

Pri analiziranju primarnih študij smo odkrili, da se ovira *b) nejasne smernice* še vedno navaja v obravnavani literaturi. Pojavlja se v primarnih študijah [SI3], [SI8] in [SI9]. V študiji [SI7] avtorji ugotavljajo, da je področje implementacije vitkega razvoja programske opreme velikih razsežnosti zelo slabo raziskano.

4.3.6 Kanban tabla

Analiza primarnih študij ne kaže bistvenega napredka v razumevanju in interpretiranju pojma Kanban tabla. V štirih študijah, [SI1], [SI2], [SI3] in [SI5] (44%) se pojem Kanban table interpretira pravilno kot element Kanban metode. V [SI6] (12%) je Kanban tabla nastala kot nadgradnja Scrum table. V štirih primarnih študijah [SI4], [SI7], [SI8], in [SI9] (44 %) pojem ni obravnavan.

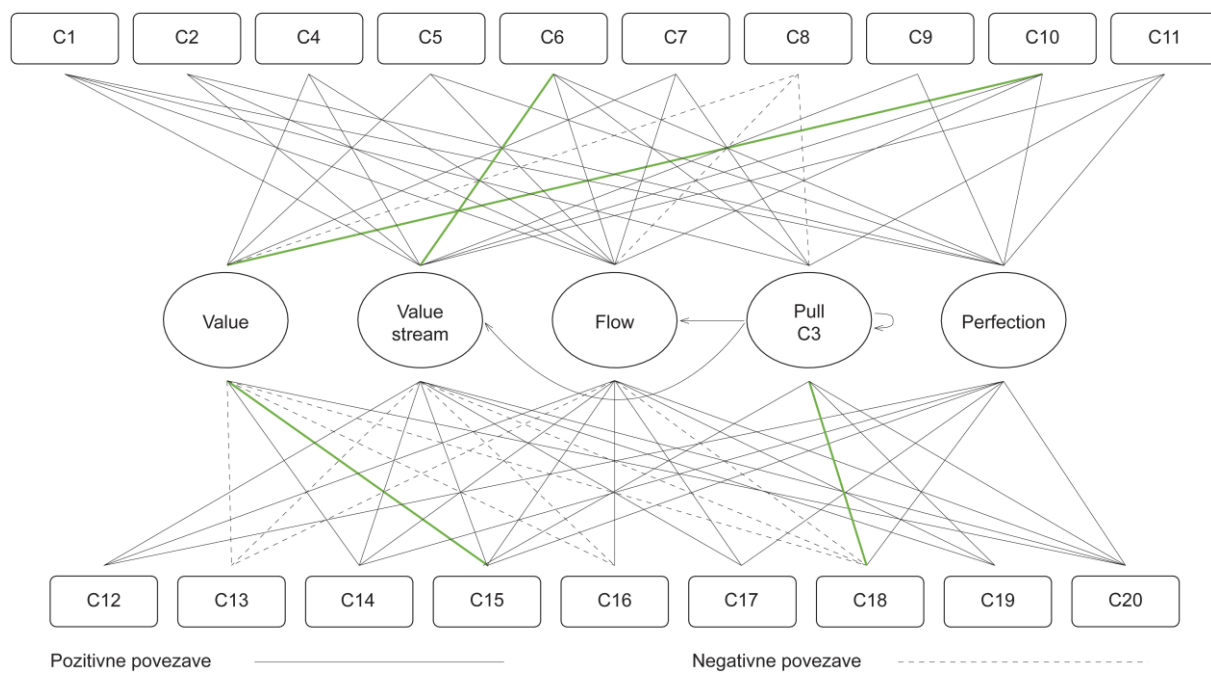
V skupini A smo pričakovali, da se bo Kanban tabla obravnavala v več študijah (obravnavala se le v eni od štirih), in da bo interpretirana kot element Kanban metode. Po drugi strani je bila v skupini D Kanban tabla pričakovano v obeh študijah obravnavana kot element Kanban metode.

4.3.7 Povezave med elementi Kanbana in načeli vitkosti

V nobeni od vključenih primarnih študij se ni obravnavalo povezav med elementi Kanban in elementi vitkosti ali manifestom agilnosti. V postopku pridobivanja podatkov smo poskušali iz primarnih študij razbrati povezave med elementi Kanbana in elementi vitkosti. Tudi v naši raziskavi smo poskušali odkriti dve vrsti povezav: pozitivne in negativne. Negativne povezave so tiste, ki so bile v vsaj eni primarni študiji navedene kot negativne.

Naj omenimo, da v primarnih študijah negativnih povezav nismo razbrali razen v [SI2] – primerjalni članek [6], kjer sta avtorja že opravila takšno analizo. Povezave med elementi, za katere sta avtorja ugotovila, da so negativne, smo kot take upoštevali tudi v našem grafu.

Slika Slika 27 prikazuje povezave med elementi Kanban pristopa in načeli vitkosti, ki so nastale na osnovi analize vključenih primarnih študij v našem sistematičnem pregledu. Da je primerjava z grafom na sliki Slika 17 lažja, so na spodnji sliki povezave, ki so bile dodatno odkrite ob analizi vključenih primarnih študij v okviru naše raziskave, označene z zeleno črto.



Slika 27: Povezave med elementi Kanban pristopa in načeli vitkosti.

Poglavje 5 Sklepne ugotovitve

Eden od glavnih ciljev naloge je bil primerjati ugotovitve sistematičnega pregleda s tistimi iz *The kanban approach, between agility and leanness: a systematic review* [6], v katerem se je obravnavala literatura do leta 2012. V diplomskem delu smo poskušali članek [6] nadgraditi tako, da smo preučili literaturo, ki je bila objavljena od leta 2013 dalje.

V sistematičnem pregledu literature smo obravnavali 9 primarnih študij. Študije so bile empirične narave, njihov kontekst pa je bil zelo različen – od manjših projektov do procesov izgradnje programske opreme velikih razsežnosti. Nobena od obravnavanih primarnih študij ni bila osredotočena le na implementacijo Kanbana, ampak je obravnavala neko širšo tematiko. Podobno, kot sta že Al-Baik in Miller ugotavljala v [6], se tudi v primarnih študijah, ki smo jih obravnavali v tej nalogi, Kanban najpogosteje obravnava kot orodje vitkega razvoja (44,4%). V štirih študijah se beseda kanban ne uporablja za pojmovanje Kanbana kot metode, kot jo je definira Anderson v [2], ampak se nanaša na samostojno agilno metodologijo, ali se njen pomen celo zamenja s kanban sistemom.

V obravnavanih primarnih študijah smo se srečali z vsemi koristmi in izzivi, ki sta jih Al-Baik in Miller ugotavljala že v [6]. Poleg teh pa smo odkrili tudi nove. Iz rezultatov lahko sklepamo, da so v primarnih študijah več poročani pozitivni učinki kot pa izzivi implementacije Kanban sistema. To lahko razlagamo na dva načina. Prvi je ta, da uvedba Kanbana v splošnem prinaša več koristi za organizacijo kot pa težav pri njegovem implementiranju. Z drugimi besedami bi lahko rekli, da je prehod na uporabo Kanbana priporočljiv. Druga interpretacija pa je lahko ta, da so bili avtorji v svojih raziskavah nagnjeni bolj k raziskovanju pozitivnih učinkov Kanbana.

Kot glavne koristi vpeljave Kanbana primarne študije poročajo: povečuje zadovoljstvo kupca in omogoča ustvarjanje izdelkov visokega cenovnega razreda; povečuje predvidljivost izdaje novih izdelkov kupcu z upoštevanjem spreminjanja kupčevih zahtev; zagotavlja razvoj znanja in spodbuja povezanost članov ekip; ter zmanjšuje stroške in povečuje propustnost in učinkovitost procesa razvoja programske opreme. Vse našete prednosti vpeljave Kanbana so bile navajane v 55,6% obravnavanih primarnih študij.

Kot glavni izziv se, enako kot v [6], še vedno navaja pomanjkljivo določene smernice za vpeljavo Kanban pristopa in njegovih elementov. To navaja kar 44,4% obravnavanih študij. Sledijo mu težave z vpeljevanjem nekaterih elementov Kanban pristopa, kot je npr. vzpostavljanje zveznega toka. (angl. flow). Te navaja 33,3% primarnih študij.

V obravnavi smernic za uvedbo Kanbana in njegovih elementov ni opaziti bistvene razlike v primerjavi z ugotovitvami v [6]. Ugotovili smo rahel porast obravnave elementov Kanbana v obravnavani literaturi v primerjavi s sistematičnim pregledom v [6]. Konkretnih smernic za uvedbo Kanbana v organizacijo primarne študije, razen [6], ki je bila tudi obravnavana med njimi, niso podajale.

V zaključku članka *The kanban approach, between agility and leanness: a systematic review* [6] avtorja navedeta, da je raziskanost področja vpeljave Kanbana v proces razvoja programske opreme zelo slaba in, da se kaže velika potreba po sistematičnih študijah o koristih, ki jih v organizacijo prinašata vitkost in Kanban. Pri tem avtorja mislita na to, kaj je treba v trenutno uveljavljenih interpretacijah vitkosti in Kanbana spremeniti, da bosta bolj prilagojena za področje razvoja programske opreme in IT industrije na splošno. Čeprav smo v naši raziskavi ugotovili, da se na tem področju nedvomno kažejo napredki, tudi v zaključku tega dela povdarjamo, da je še naprej potrebno nadaljevati z raziskavami, ki bodo praktikom omogočala lažji prehod na uporabo Kanban metode v svojih organizacijah.

Seznam slik

Slika 1: Stanje uporabe agilnih metodologij na področju razvoja programske opreme v letu 2015 [5].	2
Slika 2: Primer kanban kartice v sistemu TPS [8].	8
Slika 3: Primer vizualizacije delovnega toka z diagramom stanj [2].	10
Slika 4: Primer Kanban table [2].	11
Slika 5: Primer prikaza delovne naloge s fiksnim rokom izvedbe na post-it listku [2].	13
Slika 6: Primer drevesnega diagrama za prikaz kvantitativnih rezultatov [7].	34
Slika 7: Primer lijakastega prikaza [7].	36
Slika 8: Postopek izbire primarnih študij.	45
Slika 9: Orodje za preverjanje kakovosti študij [6].	47
Slika 10: Porazdelitev primarnih študij glede na vrsto gradiva [6].	49
Slika 11: Porazdelitev primarnih študij po skupinah pojmovanja Kanban in po tipu študije.	50
Slika 12: Porazdelitev primarnih študij po skupinah pojmovanja Kanban in letu objave.	51
Slika 13: Obravnava elementov Kanbana v primarnih študijah tipa A [6].	52
Slika 14: Obravnava elementov Kanbana v primarnih študijah tipa B [6].	54
Slika 15: Obravnava elementov Kanbana v primarnih študijah tipa C [6].	55
Slika 16: Obravnava elementov Kanbana v primarnih študijah tipa D [6].	56
Slika 17: Povezave med elementi Kanban pristopa in načeli vitkosti [6].	60
Slika 18: Orodje za ocenjevanje kakovosti primarnih študij v sistematičnem pregledu literature.	64
Slika 19: Porazdelitev primarnih študij glede na vrsto gradiva.	66
Slika 20: Porazdelitev primarnih študij po skupinah pojmovanja Kanban in po tipu študije.	66
Slika 21: Prikaz primarnih študij po skupinah pojmovanja Kanbana in letu objave.	67
Slika 22: Obravnava elementov Kanbana v primarnih študijah tipa A.	69
Slika 23: Obravnava elementov Kanbana v primarnih študijah tipa B.	70
Slika 24: Obravnava elementov Kanbana v primarni študiji tipa C.	72
Slika 25: Obravnava elementov Kanbana v primarnih študijah tipa D.	74
Slika 26: Primerjava pogostosti obravnave elementov Kanbana v literaturi glede na sistematični pregled.	74
Slika 27: Povezave med elementi Kanban pristopa in načeli vitkosti.	81

Seznam tabel

Tabela 1: Faze, aktivnosti in izdelki sistematičnega pregleda literature.....	41
Tabela 2: Iskalni nizi.....	44
Tabela 3: Skupina glede na obravnavo Kanban v primarnih študijah.	50
Tabela 4: Seznam koristi vpeljave Kanban pristopa v organizacijo.	57
Tabela 5: Seznam izzivov pri vpeljavi Kanban pristopa v organizacijo.....	58
Tabela 6: Koristi, ugotovljene v članku [6], ki so bile obravnavane v primarnih študijah.....	76
Tabela 7: Izzivi, ugotovljeni v članku [6], ki so bile obravnavane v primarnih študijah.	76
Tabela 8: Koristi vpeljave Kanbana, ki niso bile obravnavane v članku [6].	77
Tabela 9: Izzivi vpeljave Kanbana, ki niso bile obravnavane v članku [6].	77

Literatura

- [1] J. Womack, D. T. Jones in D. Roos, *The Machine That Changed The World*, New York: HarperPerennial, 1990.
- [2] D. J. Anderson, *Kanban: Successful Evolutionary Change for Your Technology Business*, Washington: Blue Hole Press, 2010.
- [3] VersionOne, Inc., „3rd Annual State Of Agile Survey: "The State Of Agile Development",“ VersionOne, Inc., Alpharetta, GA, 2008.
- [4] VersionOne, „6th Annual State Of Agile Survey: "The State Of Agile Development",“ VersionOne, 2012.
- [5] VersionOne, „10th Annual State Of Agile Survey: "The State Of Agile Development",“ VersionOne, 2016.
- [6] O. Al-Baik in J. Miller, „The kanban approach, between agility and leanness: a systematic review,“ *Empirical Software Engineering*, Izv. 20, št. 6, pp. 1861-1897, 2015.
- [7] B. A. Kitchenham in S. Charters, „Guidelines for performing Systematic Literature Reviews in Software Engineering Ver.2.3,“ EBSE Technical Report EBSE-2007-01, UK Economics and Physical Sciences Research Council, 2007.
- [8] J. P. Womack in D. T. Jones, *Lean Thinking: Banish Waste and Create Wealth in Your Corporation*, New York: Free Press, 2003.
- [9] K. e. a. BECK, „Manifesto for Agile Software Development,“ 2001. [Elektronski]. Available: <http://agilemanifesto.org/>.
- [10] P. Freeman, „Lean concepts in software engineering,“ v *IPSS-Europe International Conference on Lean Software Development*, Stuttgart, Nemčija, 1992.
- [11] P. Rodríguez, J. Partanen, P. Kuvaja in M. Oivo, „Combining Lean Thinking and Agile Methods for Software Development: A Case Study of a Finnish Provider of Wireless Embedded Systems,“ v *Proceedings of the Annual Hawaii International*

Conference on System Sciences, Waikoloa, Združene države Amerike, 2014.

- [12] M. Poppendieck in T. Poppendieck, *Lean Software Development: An Agile Toolkit*, Boston: Addison Wesley, 2003.
- [13] S. Shingo, *Studi of Toyota Production System*, Tokyo, Japonska: Japan Management Association, 1981.
- [14] D. J. A. & A. Inc., „David J. Anderson & Associates Inc.,“ [Elektronski]. Available: <http://www.djaa.com/principles-kanban-method-0>. [Poskus dostopa 17 07 2016].
- [15] W. A. Shewhart, *Economic control of quality of manufactured product*, New York: D. Van Nostrand Company, Inc., 1931.
- [16] T. Dybå in T. Dingsøy, „Empirical studies of agile software development: a systematic review,“ *Information and Software Technology*, Izv. 50, št. 9 - 10, pp. 833 - 859, 2008.
- [17] X. Wang, K. Conboy in O. Cawley, „"Leagile" software development: An experience report analysis of the application of lean approaches in agile software development,“ *Journal of Systems and Software*, Izv. 85, št. 6, pp. 1287 - 1299, 2012.
- [18] J. L. Jensen in R. Rodgers, „Cumulating the intellectual gold of case study research,“ *Public Administration Review*, Izv. 61, št. 2, pp. 235 - 246, 2001.
- [19] International Institute of Business Analysis, „A Guide to the Business Analysis Body of Knowledge (BABOK® Guide) Version 2.0,“ International Institute of Business Analysis, Toronto, Ontario, Canada, 2009.
- [20] N. Kano, *Guide to TQM in service industries*, Tokyo: Asian Productivity Organization, 1996.
- [21] M. Staron, W. Meding in B. Söderqvist, „A method for forecasting defectbacklog in large streamline software development projects and its industrial evaluation,“ *Information and Software Technology*, Izv. 52, št. 10, pp. 1069 - 1079, 2010.

- [22] K. Petersen in C. Wohlin, „Software process improvement through theLean Measurement (SPILEAM) method,“ *Journal of Systems and Software*, Izv. 83, št. 7, p. 1275–1287, 2010.
- [23] J. M. Morgan in J. K. Liker, „The Toyota Product Development System: Integrating People, Process and Technology,“ *Journal of Product Innovation Management*, Izv. 24, št. 3, p. 191–281, 2007.
- [24] T. I. Corporation, *Toyota Production System and what it means for business*, Dentsu Brussels Group, 2010.

Priloge

Priloga I – Predlog oblike poročila sistematičnega pregleda literature

Poglavje	Podpoglavje	Okvir	Komentar
Naslov*			Naslov naj bo kratek in jedrnat. Temelji naj na raziskovalnem vprašanju. V objavah v revijah naj nakazuje, da je študija sistematični pregled literature.
Avtorstvo*			Ko raziskavo izvaja več raziskovalcev, naj bo avtorstvo in soavtorstvo vnaprej določeno. Prispevki raziskovalcev, ki ne bodo navedeni kot avtorji, naj bodo omenjeni v zahvalah.
Povzetek*	Kontekst	Določa pomembnost raziskovalnih vprašanj, ki so obdelana v pregledu.	Povzetek naj bralcu omogoči hitro oceno relevantnosti, kakovosti in splošnosti sistematičnega pregleda.
	Cilji	Vprašanja, ki so obdelana v pregledu.	
	Metode	Viri podatkov, izbira gradiva, orodje za oceno kakovosti in pridobivanje podatkov.	

	Rezultati	Glavne ugotovitve, vključno z rezultati analiz in analiz občutljivosti.	
	Zaključek	Predlogi za obravnavo ugotovitev v praksi in nadaljnje raziskave.	
Ozadje		Utemeljitev za nastanek pregleda, povzetek predhodnih pregledov.	Opis tehnike, ki je predmet raziskovanja, in njena predvidena pomembnost.
Vprašanja pregleda		Navedena naj bodo vsa vprašanja pregleda.	Določiti primarna in sekundarna vprašanja pregleda. Vsebina tega poglavja je lahko vključena v poglavje ozadje.
Metode pregleda	Viri podatkov in strategija iskanja.		Ta vsebina naj bo osnovana na podlagi protokola pregleda. Vsaka sprememba prvotno določenega raziskovalnega protokola naj bo zabeležena.
	Izbira študij		
	Ocena kakovosti študij		
	Pridobivanje podatkov		

Povzetek podatkov			
Vključene in izključene študije		Kriteriji za vključitev in izključitev. Seznam izključenih študij z vzrokom za izključitev.	Kriterije za vključitev in izključitev študij lahko prikažemo z diagramom poteka, saj se lahko študije izključi iz obravnave v različnih korakih pregleda in iz različnih razlogov.
Rezultati	Ugotovitve	Opis primarnih študij, rezultati kvantitativnih povzetkov, podrobnosti analiz.	Za vsako od študij naj bodo navedeni kvalitativni povzetki v tabelarni obliki. Rezultati kvantitativnih povzetkov naj bodo predstavljeni v tabelah in grafih.
	Analiza občutljivosti		
Razprava	Glavne ugotovitve		Te se morajo ujemati z ugotovitvami, ki so navedene v poglavju rezultati.
	Prednosti in slabosti	Prednosti in slabosti dokazanega v pregledu. Primerjava z drugimi pregledi s poudarkom na razlikah v kvaliteti in rezultatih.	Razprava o veljavnosti dokazov z upoštevanjem pristranskosti v sistematičnem pregledu omogoča bralcu, da oceni zanesljivost zbranih dokazov. A discussion of the validity of the evidence considering bias in the systematic review

			allows a reader to assess the reliance that may be placed on the collected evidence
	Pomen ugotovitev	<p>Smer in razsežnosti učinkov, ki so bili opazovani v študijah.</p> <p>Ocenitev uporabnosti ugotovitev.</p>	<p>Preveriti v kakšni meri napeljujejo rezultati na vzročnost, tako da se obravnava raven dokazov.</p> <p>Obravnavati prednosti, negativne učinke in tveganja.</p> <p>Obravnavati spremenljivost učinkov in njen vzrok.</p>
Zaključek	Priporočila	<p>Praktične posledice za razvoj programske opreme.</p> <p>Neodgovorjena vprašanja in priporočila za nadaljnje raziskave.</p>	Kako vplivajo posledice rezultatov pregleda na delovanje v praksi.
Zahvale*		Navede se vse, ki so prispevali k raziskavi in niso navedeni kot avtorji ali soavtorji.	
Konflikt interesov			Vsi postranski interesi raziskovalcev (npr. finančni) morajo biti navedeni.
Viri in priloge			V prilogah lahko navedemo seznam vključenih in izključenih študij, navedemo

podrobnosti strategije iskanja, in za navedbo surovih podatkov (angl. raw data), ki smo jih pridobili iz študij.

Priloga II – Kodirna shema

Kategorija	Opis	Kodirna pravila
Metoda Kanban	Metoda inkrementalnega in razvijalnega spreminjanja procesa znotraj organizacije.	Opis opravljenih korakov za vpeljavo Kanban pristopa v organizacijo.
Kanban tabla	Vizualno orodje za prikaz in optimizacijo delovnega procesa, ki temeljijo na Kanban. Njihova naloga je tudi spodbujanje ekipnega dela in sodelovanja med člani ekipe.	Navedba kakršnega koli orodja, ki se uporablja za vizualizacijo opravil ali delovnega procesa razvoja programske opreme.
Izvelčni sistem	Začetek dela se proži, ko se pojavi zahteva po njegovem učinku, delo se izvaja v skladu z željami in količinami, ki jih kupec zahteva.	Pristop, pri katerem se proces razvoja izdelka ali storitve začne ob nastanku zahteve s strani kupca.
Delovna naloga	Naloga, ki je vizualizirana na Kanban tabli.	Zahteva, uporabniška zgodba, ali katerikoli predmet, ki je treba vizualizirati, in se pomika skozi različne faze razvojnega procesa.
Razvrščanje po prioriteti	Urejen seznam zahtev ali delovnih nalog.	Katerikoli izraz, ki nakazuje uporabo orodja za delo z vrstami, vrstnim redom ali razvrščanjem po prioriteti.
Kriteriji vključitve	Pravila po katerih se odloča ali se delovna naloga doda na orodje za vizualizacijo. Zaželeno je, da	Norma, pravilo ali standard, ki določa katere delovne naloge se vključijo in postavijo na tablo.

pravila upoštevajo dodano vrednost za kupca.

Omejitev obsega dela v teku (WIP)	Število delovnih nalog, ki so lahko sočasno v izvajanju v posamezni fazi razvojnega procesa.	Pravilo, tehnika za določanje števila sočasno izvajanih delovnih predmetov.
Zaključena delovna naloga	Delovne naloge, ki so bile v neki fazi delovnega procesa zaključene in so pripravljene za izvedbo v naslednji fazi delovnega procesa.	Delovne naloge, ki so bile zaključene v nekem razvojnem procesu in jih lahko naslednji razvojni proces "izvleče".
Vzratna delovna naloga	Delovne naloge, ki se po Kanban tabli pomikajo v vzratni smeri.	Delovna naloga, ki se mora vrniti v predhodni delovni proces. Na primer: odpravljanje napake.
Vrednostno učenje	Proces učenja v katerem se odkriva kaj je za organizacijo dobro in kaj ne, z namenom vrednotenja s poslovnega vidika.	Z vrednostnim učenjem organizacija s časom pridobi znanje za pridobivanje kupcev, ustrežanju njihovim zahtevam in prodajanju izdelkov oz. storitev kupcem. Na vrednostno učenje lahko gledamo tudi kot na PDCA cikel – planiranje, izvedba, preverjanje in prilagajanje (angl. Plan, Do, Check, Adjust).
Čas cikla	Dejanski čas, ki je bil porabljen za končanje delovne naloge brez upoštevanja časa, ko je delovna naloga prebila v čakalnih in izravnalnih vrstah.	Čas cikla je enak času, ki je bil porabljen za izdelavo delovne naloge brez zakasnitev, od začetka opravljanja delovne naloge, do dostave kupcu.
Potreben čas	Celotni čas, ki preteče od kupčeve zahteve po izdelku oz. storitvi do dostave izdelka oz. storitve	Čas od pojava zahteve po izdelku oz. storitvi do

	kupcu.	trenutka dostave izdelka oz. storitve kupcu.
Orodje za merjenje	Navadno gre za orodje za izdelavo diagramov, ki prikazuje pot delovne naloge skozi delovni proces.	Katerikoli merilni diagram, ki se uporablja za predstavitev zmogljivosti Kanban.
Ozko grlo	Ozka grla so omejitve v delovnem toku, ki zmanjšujejo pretočnost sistema.	Delovne naloge, ki upočasnjujejo delovni tok.
Čakalne vrste	Načrtovan čas v delovnem procesu, katerega naloga je ohranjanje konstantnega časa predaje izdelkov.	Čas, ki je bil rezerviran z namenom ohranjanja konstantnega časa predaje izdelkov.
Načela Kanbana	Načela, ki so kot ogrodje za vpeljavo Kanban pristopa v proces razvoja programske opreme.	To so temeljna načela Kanban pristopa.
Tehnike Kanban	Sistem za upravljanje delovnega procesa, ki se uporablja za odpravljanje težav.	Tehnike, ki ne dovoljujejo izkoriščanja delovne sile in zaposlenim omogočajo odrekanje dela, če so zahteve nemogoče.
Avatar	Vizualna predstavitev podrobnosti delovne naloge.	Vizualna predstavitev podrobnosti delovne naloge. To vključuje beležne lističe, simbole, idr.

Priloga III – Seznam virov literature s podrobnostmi izvedbe iskanja

1. Pregled spletnih virov in parametrov iskanja

Spletni viri	naslov	Iskalni niz za monografije	Iskalni niz za sistematične preglede	Datum poizvedb	Št. zadetkov Mon./sist.p.	Opombe
Scopus						
https://www.scopus.com/		(TITLE-ABS-KEY ("Software engineering" OR "Software development" OR "Information technology" OR "Computer science" OR "IT project management" OR "Software project management" OR "Information technology project management" OR "Software product development" OR "IT Product development") AND TITLE-ABS-KEY (kanban OR lean OR "Continuous flow" OR "Pull system" OR "Lead	(TITLE-ABS-KEY ("Software engineering" OR "Software development" OR "Information technology" OR "Computer science" OR "is project management" OR "Software project management" OR "Information technology project management" OR "Software product development" OR "IT Product development") AND TITLE-ABS-KEY (kansan OR lean OR "Continuous flow" OR "Pull	16.5.2016	8 / 1	

	time" OR "Cycle time" OR "Work in progress" OR "Work in process" OR wip) AND TITLE-ABS-KEY (agile OR agility OR scrum OR leagile OR scrumban) AND TITLE-ABS-KEY ("Process improvement" OR "Cost reduction" OR "Reducing Cost" OR "Reduce cost" OR "High quality" OR "Higher quality" OR "Quality improvement" OR "Improving quality" OR "Improve quality") AND TITLE-ABS-KEY (empirical OR experiment OR "Evidence based" OR "Industrial setting" OR "Case study" OR "Action research")) AND PUBYEAR > 2012 AND LANGUAGE (english)	system" OR "Lead time" OR "Cycle time" OR "Work in progress" OR "Work in process" OR wip) AND TITLE-ABS-KEY (agile OR agility OR scrum OR fragile OR schuman) AND TITLE-ABS-KEY ("Process improvement" OR "Cost reduction" OR "Reducing Cost" OR "reduced cost" OR "High quality" OR "Higher quality" OR "Quality improvement" OR "Improving quality" OR "improved quality") AND TITLE-ABS-KEY ("Literature Review" OR overview OR "Research review" OR "Research synthesis" OR "Research integration" OR "Systematic review" OR "Integrative			
--	--	---	--	--	--

		research review" OR "Integrative review")) AND PUBYEAR > 2012 AND LANGUAGE (english)			
ACM Digital Library					
http://dl.acm.org/	"query": { ("Software engineering" "Software development" "Information technology" "Computer science" "IT project management" "Software project management" "Information technology project management" "Software product development" "IT Product development") +(Kanban Lean "Continuous flow" "Pull system" "Lead time" "Cycle time" "Work in progress" "Work in process" WIP) +(Agile Agility Scrum Leagile Scrumban) +("Process	"query": { ("Software engineering" "Software development" "Information technology" "Computer science" "IT project management" "Software project management" "Information technology project management" "Software product development" "IT Product development") +(Kanban Lean "Continuous flow" "Pull system" "Lead time" "Cycle time" "Work in progress" "Work in process" WIP) +(Agile Agility Scrum Leagile Scrumban)	17.5.2016	13 / 2	Za podatkovno bazo je določen: The ACM Guide to Computing Literature.

	improvement" "Cost reduction" "Reducing Cost" "Reduce cost" "High quality" "Higher quality" "Quality improvement" "Improving quality" "Improve quality") +(Empirical Experiment "Evidence based" "Industrial setting" "Case study" "Action research") } "filter": {"publicationYear":{ "gte":2013, "lte":2016 }}, {owners.owner=GUIDE}	+("Process improvement" "Cost reduction" "Reducing Cost" "Reduce cost" "High quality" "Higher quality" "Quality improvement" "Improving quality" "Improve quality") +("Literature Review" Overview "Research review" "Research synthesis" "Research integration" "Systematic review" "Integrative research review" "Integrative review") } "filter": {"publicationYear":{ "gte":2013, "lte":2016 }}, {owners.owner=GUIDE}			
IEEE Xplore					
http://ieeexplore.ieee.org/search/advs	((("Software development" OR "Project management" OR	((("Software development" OR "Project management"	17.5.2016	8 / 0	Prvotnega iskalnega niza ni možno

earch.jsp	"Information technology" OR "Product development" OR IT) AND (Kanban OR Lean) AND (Agil* OR Scrum* OR Leagile) AND ("Process improvement" OR cost OR quality) AND (Empirical OR Experiment OR "Case study" OR "Industrial setting"))	OR "Information technology" OR "Product development" OR IT) AND (Kanban OR Lean) AND (Agil* OR Scrum* OR Leagile) AND ("Process improvement" OR cost OR quality) AND (Review OR "Research synthesis" OR "Research integration"))			uporabiti, saj je iskalnik omejen na 15 iskalnih parametrov. Iskalna niza sta bila prilagojena! Za podrobnosti glej točko 2 te priloge. Iskanja so bila izvedena v načinu Advanced Search, Command Search, Metadata Only
Science Direct					
http://www.science-direct.com/science?_ob=MiamiSearchURL&_method=requestForm&_temp=all_boolSearch.tmpl&md5=052b06d957a9d8c82e07acf1d7eef1b7	pub-date > 2012 and TITLE-ABSTR-KEY("Software engineering" OR "Software development" OR "Information technology" OR "Computer science" OR "IT project management" OR "Software project management" OR "Information technology project management" OR "Software product	pub-date > 2012 and TITLE-ABSTR-KEY("Software engineering" OR "Software development" OR "Information technology" OR "Computer science" OR "IT project management" OR "Software project management" OR "Information technology project management" OR	17.5.2016	1 / 1	

	development" OR "IT Product development") AND TITLE-ABSTR-KEY(Kanban OR Lean OR "Continuous flow" OR "Pull system" OR "Lead time" OR "Cycle time" OR "Work in progress" OR "Work in process" OR WIP) AND TITLE-ABSTR-KEY(Agile OR Agility OR Scrum OR Leagile OR Scrumban) AND TITLE-ABSTR-KEY("Process improvement" OR "Cost reduction" OR "Reducing Cost" OR "Reduce cost" OR "High quality" OR "Higher quality" OR "Quality improvement" OR "Improving quality" OR "Improve quality") AND TITLE-ABSTR-KEY(Empirical OR Experiment OR "Evidence based" OR "Industrial setting" OR "Case study" OR "Action	"Software product development" OR "IT Product development") AND TITLE-ABSTR-KEY(Kanban OR Lean OR "Continuous flow" OR "Pull system" OR "Lead time" OR "Cycle time" OR "Work in progress" OR "Work in process" OR WIP) AND TITLE-ABSTR-KEY(Agile OR Agility OR Scrum OR Leagile OR Scrumban) AND TITLE-ABSTR-KEY("Process improvement" OR "Cost reduction" OR "Reducing Cost" OR "Reduce cost" OR "High quality" OR "Higher quality" OR "Quality improvement" OR "Improving quality" OR "Improve quality") AND TITLE-ABSTR-KEY("Literature Review" OR			
--	---	---	--	--	--

	research")	Overview OR "Research review" OR "Research synthesis" OR "Research integration" OR "Systematic review" OR "Integrative research review" OR "Integrative review")			
Web of Science					
http://webofscience.com	TOPIC: ("Software engineering" OR "Software development" OR "Information technology" OR "Computer science" OR "IT project management" OR "Software project management" OR "Information technology project management" OR "Software product development" OR "IT Product development") AND TOPIC: (Kanban OR Lean OR "Continuous flow" OR "Pull system" OR "Lead time" OR	TOPIC: ("Software engineering" OR "Software development" OR "Information technology" OR "Computer science" OR "IT project management" OR "Software project management" OR "Information technology project management" OR "Software product development" OR "IT Product development") AND TOPIC: (Kanban OR Lean OR "Continuous flow" OR	17.5.2016	4 / 1	

	<p>"Cycle time" OR "Work in progress" OR "Work in process" OR WIP) AND TOPIC: (Agile OR Agility OR Scrum OR Leagile OR Scrumban) AND TOPIC: ("Process improvement" OR "Cost reduction" OR "Reducing Cost" OR "Reduce cost" OR "High quality" OR "Higher quality" OR "Quality improvement" OR "Improving quality" OR "Improve quality") AND TOPIC: (Empirical OR Experiment OR "Evidence based" OR "Industrial setting" OR "Case study" OR "Action research")</p> <p>Timespan: 2013-2016.</p> <p>Search language=English</p>	<p>"Pull system" OR "Lead time" OR "Cycle time" OR "Work in progress" OR "Work in process" OR WIP) AND TOPIC: (Agile OR Agility OR Scrum OR Leagile OR Scrumban) AND TOPIC: ("Process improvement" OR "Cost reduction" OR "Reducing Cost" OR "Reduce cost" OR "High quality" OR "Higher quality" OR "Quality improvement" OR "Improving quality" OR "Improve quality") AND TOPIC: ("Literature Review" OR Overview OR "Research review" OR "Research synthesis" OR "Research integration" OR "Systematic review" OR "Integrative research review" OR</p>			
--	---	--	--	--	--

		"Integrative review") Timespan: 2013-2016. Search language=English			
InterScience (Wiley)					
http://onlinelibrary.wiley.com/	1. Article Title ("Software engineering" OR "Software development" OR "Information technology" OR "Computer science" OR "IT project management" OR "Software project management" OR "Information technology project management" OR "Software product development" OR "IT Product development") in Article Titles AND (Kanban OR Lean OR "Continuous flow" OR "Pull system" OR "Lead time" OR "Cycle time" OR "Work in progress" OR	1. Article Title ("Software engineering" OR "Software development" OR "Information technology" OR "Computer science" OR "IT project management" OR "Software project management" OR "Information technology project management" OR "Software product development" OR "IT Product development") in Article Titles AND (Kanban OR Lean OR "Continuous flow" OR "Pull system" OR "Lead time" OR "Cycle time"	17.5.2016	1. Article Title 0 / 0 2. Keywords 0 / 0 3. Abstract 1 / 1	Ker iskalnik ne omogoča hkratnega iskanja po Title, Keyword, Abstract, je bilo treba izvesti tri ločene poizvedbe za vsak tip literature (monographic, systematic lit.).

	<p>"Work in process" OR WIP) in Article Titles AND (Agile OR Agility OR Scrum OR Leagile OR Scrumban) in Article Titles AND ("Process improvement" OR "Cost reduction" OR "Reducing Cost" OR "Reduce cost" OR "High quality" OR "Higher quality" OR "Quality improvement" OR "Improving quality" OR "Improve quality") in Article Titles AND (Empirical OR Experiment OR "Evidence based" OR "Industrial setting" OR "Case study" OR "Action research") in Article Titles between years 2012 and 2017</p> <p>2. Keywords</p> <p>("Software engineering" OR "Software development" OR "Information technology" OR "Computer science" OR "IT</p>	<p>OR "Work in progress" OR "Work in process" OR WIP) in Article Titles AND (Agile OR Agility OR Scrum OR Leagile OR Scrumban) in Article Titles AND ("Process improvement" OR "Cost reduction" OR "Reducing Cost" OR "Reduce cost" OR "High quality" OR "Higher quality" OR "Quality improvement" OR "Improving quality" OR "Improve quality") in Article Titles AND ("Literature Review" OR Overview OR "Research review" OR "Research synthesis" OR "Research integration" OR "Systematic review" OR "Integrative research review" OR "Integrative review") in Article Titles between years</p>			
--	--	--	--	--	--

	<p>project management" OR "Software project management" OR "Information technology project management" OR "Software product development" OR "IT Product development") in Keywords AND (Kanban OR Lean OR "Continuous flow" OR "Pull system" OR "Lead time" OR "Cycle time" OR "Work in progress" OR "Work in process" OR WIP) in Keywords AND (Agile OR Agility OR Scrum OR Leagile OR Scrumban) in Keywords AND ("Process improvement" OR "Cost reduction" OR "Reducing Cost" OR "Reduce cost" OR "High quality" OR "Higher quality" OR "Quality improvement" OR "Improving quality" OR "Improve quality") in Keywords AND (Empirical</p>	<p>2012 and 2017</p> <p>2. Keywords</p> <p>("Software engineering" OR "Software development" OR "Information technology" OR "Computer science" OR "IT project management" OR "Software project management" OR "Information technology project management" OR "Software product development" OR "IT Product development") in Keywords AND (Kanban OR Lean OR "Continuous flow" OR "Pull system" OR "Lead time" OR "Cycle time" OR "Work in progress" OR "Work in process" OR WIP) in Keywords AND (Agile OR Agility OR Scrum OR Leagile OR Scrumban) in</p>			
--	--	--	--	--	--

	<p>OR Experiment OR "Evidence based" OR "Industrial setting" OR "Case study" OR "Action research") in Keywords between years 2012 and 2017</p> <p>3. Abstract</p> <p>("Software engineering" OR "Software development" OR "Information technology" OR "Computer science" OR "IT project management" OR "Software project management" OR "Information technology project management" OR "Software product development" OR "IT Product development") in Abstract AND (Kanban OR Lean OR "Continuous flow" OR "Pull system" OR "Lead time" OR "Cycle time" OR "Work in progress" OR "Work in process" OR WIP) in</p>	<p>Keywords AND ("Process improvement" OR "Cost reduction" OR "Reducing Cost" OR "Reduce cost" OR "High quality" OR "Higher quality" OR "Quality improvement" OR "Improving quality" OR "Improve quality") in Keywords AND ("Literature Review" OR Overview OR "Research review" OR "Research synthesis" OR "Research integration" OR "Systematic review" OR "Integrative research review" OR "Integrative review") in Keywords between years 2012 and 2017</p> <p>3. Abstract</p> <p>("Software engineering" OR "Software development" OR "Information technology" OR</p>			
--	---	--	--	--	--

	<p>Abstract AND (Agile OR Agility OR Scrum OR Leagile OR Scrumban) in Abstract AND ("Process improvement" OR "Cost reduction" OR "Reducing Cost" OR "Reduce cost" OR "High quality" OR "Higher quality" OR "Quality improvement" OR "Improving quality" OR "Improve quality") in Abstract AND (Empirical OR Experiment OR "Evidence based" OR "Industrial setting" OR "Case study" OR "Action research") in Abstract between years 2012 and 2017</p>	<p>"Computer science" OR "IT project management" OR "Software project management" OR "Information technology project management" OR "Software product development" OR "IT Product development") in Abstract AND (Kanban OR Lean OR "Continuous flow" OR "Pull system" OR "Lead time" OR "Cycle time" OR "Work in progress" OR "Work in process" OR WIP) in Abstract AND (Agile OR Agility OR Scrum OR Leagile OR Scrumban) in Abstract AND ("Process improvement" OR "Cost reduction" OR "Reducing Cost" OR "Reduce cost" OR "High quality" OR "Higher quality" OR "Quality</p>			
--	--	---	--	--	--

		<p>improvement" OR "Improving quality" OR "Improve quality") in Abstract AND ("Literature Review" OR Overview OR "Research review" OR "Research synthesis" OR "Research integration" OR "Systematic review" OR "Integrative research review" OR "Integrative review") in Abstract between years 2012 and 2017</p>			
Skupaj				35 / 6	

2. Sprememba iskalnih parametrov za potrebe omejitev iskalnika IEEE Xplore

Po zgledu [7] je bil iskalni niz prilagojen skladno z metodo PICOC.

1.) Iskalni niz za iskanje monografij.

Population

Spremenjeni pogoji:

- "IT Product development" OR "Software product development" sta bila zamenjana s "Product Development",
- "Software engineering" se opusti,
- "Computer science" se opusti,
- "IT project management" OR "Software project management" OR "Information technology project management" so bili zamenjani s "project management".

Intervention

Spremenjeni pogoji:

- "Continuous flow" OR "Pull system" OR "Lead time" OR "Cycle time" OR "Work in progress" OR "Work in process" OR WIP so bili odstranjeni, ker so elementi Kanban/Lean

Comparison

Spremenjeni pogoji:

- Agile OR Agility sta bila zamenjana z Agil*
- Scrumban odstranjen, ker je upoštevan v Scrum*

Outcome

Spremenjeni pogoji:

- "Cost reduction" OR "Reducing Cost" OR "Reduce cost" so bili zamenjani s cost

- "High quality" OR "Higher quality" OR "Quality improvement" OR "Improving quality" OR "Improve quality" so bili zamenjani s quality

Context

Spremenjeni pogoji:

- "Evidence based" je bil odstranjen, ker je že pogoj Empirical
- "Action research" je bil odstranjen, ker je manj pomemben od ostalih pogojev.

2.) Iskalni niz za iskanje sistematičnih pregledov literature

Za People, Intervention, Comparison in Outcome so bili pogoji spremenjeni enako, kot za iskalni niz za iskanje monografske literature.

Context

Spremenjeni pogoji:

- "Literature Review" OR "Research review" OR "Systematic review" OR "Integrative research review" OR "Integrative review" – zamenjano z review.

3. Ročni pregled revij

Ročno je bila pregledana revija Empirical Software Engineering. Pregledane so bile izdaje od Vol.18 issue 1 do Vol.21 issue 3. Najdena sta bila 2 zadetka.

Priloga IV – Seznam vključene literature

- [SI1] M. O. Ahmad et al., "Transition of Software Maintenance Teams from Scrum to Kanban", v Proceedings of the Annual Hawaii International Conference on System Sciences, Koloa, Kauai, Združene države Amerike, mar.2016, str. 5427 – 5436.
- [SI2] O. Al-Baik in J. Miller, "The kanban approach, between agility and leanness: a systematic review", Empirical Software Engineering, vol. 20, št. 6, str. 1861 - 1897, 2015.
- [SI3] O. Al-Baik in J. Miller, "Waste Identification and Elimination in Information Technology Organizations", Empirical Software Engineering, vol. 19, št. 6, str. 2019 - 2061, 2014.
- [SI4] T. Magennis, "The economic impact of software development process choice - Cycle-time analysis and Monte Carlo simulation results", v Proceedings of the Annual Hawaii International Conference on System Sciences, Kauai, Združene države Amerike, mar. 2015, str. 5055 - 5064.
- [SI5] G. S. Matharu et al., "Empirical Study of Agile Software Development Methodologies: A Comparative Analysis", ACM SIGSOFT Software Engineering Notes, vol. 40, št. 1, str. 1 – 6, 2015.
- [SI6] H.A. Mitre in L. Bermon-Angarita, Process and productivity improvement in agile software development with process libraries: Case study, v Agile Estimation Techniques and Innovative Approaches to Software Process Improvement, R. Colomo-Palacios et al., ur. Hershey, Pennsylvania: IGI Global, 2014, pogl. 1.
- [SI7] J. Pernstål, R. Feldt in T. Gorschek, "The lean gap: A review of lean approaches to large-scale software systems development", Journal of Systems and Software, vol. 86, št. 11, str. 2797 - 2821, 2013.
- [SI8] P. Rodríguez et al., "Building lean thinking in a telecom software development organization: Strengths and challenges", v ACM International Conference Proceeding Series, San Francisco, Združene države Amerike, maj 2013, str. 98 – 107.
- [SI9] P. Rodríguez et al., "Combining Lean Thinking and Agile Methods for Software Development: A Case Study of a Finnish Provider of Wireless Embedded Systems Detailed", v Proceedings of the Annual Hawaii International Conference on System Sciences, Waikoloa, Združene države Amerike, jan. 2014, str. 4770 – 4779.

Priloga V – Seznam izključene literature

- [SE1] "22nd European Conference on Systems, Software and Services Process Improvement, EuroSPI 2015", Communications in Computer and Information Science, vol. 543, str. 1 – 366, 2015.
- [SE2] "Proceedings - 2015 Agile Conference, Agile 2015", Piscataway: IEEE, 2013.
- [SE3] N. Abdel-Hamid in A. E. Hamouda, "Lean CMMI: An Iterative and Incremental Approach to CMMI-Based Process Improvement", v Proceedings of the 2015 Agile Conference, Washington DC, Združene države Amerike, 2015, str. 65 – 70.
- [SE4] U. K. Durrani, Z. Pita in J. Richardson, "The Tetrad Influences: A Case Study of an Adaptable Software Configuration Management Process", International Journal of Strategic Decision Sciences, vol. 5, št. 2, str. 30 – 42, 2014.
- [SE5] Elahi in M. Franchetti, "A new optimization model for closed-loop supply chain networks", v Technology Management Conference (ITMC), 2014 IEEE International, Piscataway, 2014, str. 1 – 9.
- [SE6] K. Ghane, "A model and system for applying Lean Six sigma to agile software development using hybrid simulation", v Technology Management Conference (ITMC), 2014 IEEE International, Piscataway, 2014, str. 1 – 4.
- [SE7] M. Holloway in C. Nwaoha, Dictionary of Industrial Terms, Hoboken: John Wiley & Sons, 2013.
- [SE8] M. Khurum, K. Petersen in T. Gorschek, "Extending value stream mapping through waste definition beyond customer perspective", Journal of Software: Evolution and Process, vol. 26, št. 12, str. 1074 – 1105, 2014.
- [SE9] M. Lines in S. W. Ambler, Introduction to Disciplined Agile Delivery: A Small Agile Team's Journey from Scrum to Continuous Delivery, CreateSpace Independent Publishing Platform, 2015.
- [SE10] P. E. McMahon, 15 Fundamentals for Higher Performance in Software Development: Includes Discussions on CMMI, Lean Six Sigma, Agile and SEMAT's Essence Framework, New York: PEM Systems, 2014.
- [SE11] P. Ng, "Theory Based Software Engineering with the SEMAT Kernel: Preliminary Investigation and Experiences", v Proceedings of the 3rd SEMAT Workshop on General

Theories of Software Engineering, New York, Združene države Amerike, 2013, str. 13 – 20.

- [SE12] T. T. Pullan, M. Bhasi in G. Madhu, "Decision support tool for lean product and process development", Production Planning and Control, vol. 24, št. 6, str. 449 – 464, 2013.
- [SE13] M. Staron et al., "Identifying Implicit Architectural Dependencies Using Measures of Source Code Change Waves", v 2013 39th Euromicro Conference on Software Engineering and Advanced Applications, IEEE, 2013, str. 325 – 332.
- [SE14] J. Stettina in E. Kroon, "Is there an agile handover? An empirical study of documentation and project handover practices across agile software teams", v Engineering, Technology and Innovation (ICE) & IEEE International Technology Management Conference, 2013 International Conference on, Piscataway, 2013, str. 1 – 12.
- [SE15] U. Viswanath, "Lean Transformation: How Lean Helped to Achieve Quality, Cost and Schedule: Case Study in a Multi Location Product Development Team", v 2014 IEEE 9th International Conference on Global Software Engineering, Piscataway, 2014, str. 95 – 99.
- [SE16] J. Whittle, "How Much Participation is Enough?: A Comparison of Six Participatory Design Projects in Terms of Outcomes", v Proceedings of the 13th Participatory Design Conference: Research Papers - Volume 1, New York, Združene države Amerike, 2014, str. 121 – 130.

Priloga VI – Predloga obrazca za pridobivanje podatkov

Id primarne študije	
Naslov primarne študije	
Avtorji	
Leto objave	
Datum pridobivanja podatkov	

1. Vrsta študije

Članek v reviji	Članek na spletu	Disertacija	Zbornik konference	Knjiga

2. Skupina glede na obravnavo Kanban v primarnih študijah

A	B	C	D

3. Raziskovalna vprašanja

3.1. RQ 1. Kateri elementi Kanban pristopa so v literaturi obravnavani kot za organizacijo koristni?

3.1.1. RQ 1.1 Elementi Kanbana

C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14	C15	C16	C17	C18	C19	C20

Element Kanbana	Opombe
C1	
C2	
C3	
C4	
C5	
C6	
C7	
C8	
C9	
C10	
C11	
C12	
C13	
C14	
C15	
C16	
C17	
C18	
C19	
C20	

3.1.2. RQ 1.2 Povezave med elementi Kanban pristopa in načeli vitkosti

	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14	C15	C16	C17	C18	C19	C20
L1																				
L2																				
L3																				
L4																				
L5																				

Legenda:

+ Pozitivna povezava
- Negativna povezava

L1 – Value
L2 – Value stream

L3 – Flow
L4 – Pull

L5 – Perfection

3.2. RQ 2 Kakšne so koristi in izzivi uporabe Kanban pristopa glede na obstoječo literaturo?

3.2.1. RQ 2.1 Koristi vpeljave Kanban pristopa

Izboljšuje vizualni nadzor in s tem podpira in olajša sprejemanje odločitev.	
Olajša usklajevanje večfunkcijskega ekipnega dela in uvaja samoorganizacijo.	
Spodbuja vpeljavo ciklov kakovosti in opravi, ki zagotavljajo kaizen (postopek nenehnega izboljševanja).	
Zmanjšuje čas cikla, oz. potrebni čas.	
Povečuje zadovoljstvo kupca in omogoča ustvarjanje izdelkov visokega cenovnega razreda.	
Zmanjšuje tehnična tveganja in tveganja, ki jih prinaša tržišče.	
Spodbuja razvoj strategij nenehnega izboljševanja.	
Povečuje predvidljivost izdaje novih izdelkov kupcu z upoštevanjem spreminjanja kupčevih zahtev.	
Zagotavlja razvoj znanja in spodbuja povezanost članov ekip.	
Usmerja in olajšuje upravljanje z organizacijskimi spremembami in spremembami organizacijske kulture.	
Povečuje kakovost izdelkov, kar se kaže v zmanjšanju števila okvar, povečanju števila izdelkov, ki ustrezajo kakovostnim merilom, in zmanjšanju števila napak.	

Drugo:

3.2.2. RQ 2.2 Izzivi vpeljave Kanban pristopa

Težko je določiti ustrezne metrike. Na primer potrebni čas (angl. lead time) se je za merjenje performans izkazal kot težaven.	
Vpeljava Kanban pristopa zahteva temeljito poznavanje konceptov, načel in praks vitkosti.	
Uporabljajo se nejasne definicije osnovnih elementov Kanban, kot na primer definicija zaključene delovne naloge.	
Ni ustreznih smernic za vpeljavo pristopa in elementov Kanban.	
IT organizacije s težavo vpeljujejo nekatere elemente Kanban pristopa, kot je npr. vzpostavljanje zveznega toka. (angl. flow).	
Treba je spremeniti organizacijo in strukturo ekip razvijalcev.	
Vpeljava Kanban pristopa v proces razvoja programske opreme in njegova integracija z	

obstoječimi agilnimi tehnikami je zapletena, draga in časovno potratna.	
Težko je določiti ustrezne metrike. Na primer potrebni čas (angl. lead time) se je za merjenje performans izkazal kot težaven.	

Drugo:

3.3. RQ 3 Jasnost obstoječih smernic za vpeljavo Kanban pristopa v organizacijo

Obravnavan korak vpeljave	Da	Ne	Če da, na kakšen način?
Prvi korak: vizualizacija delovnega toka.			
Drugi korak: Omejitev obsega dela v teku.			
Tretji korak: Merjenje in upravljanje delovnega toka.			

Drugo:

3.3.1. RQ 3.1 Ali se v študiji razlikuje med Kanban in kanban?

DA	
NE	

Opombe:

3.4. RQ 4. Ali se v obstoječi literaturi za elemente Kanban pristopa le ponovno uporabljeni agilni pojmi?

3.4.1. RQ.4.1 Kanban tabla

Interpretacija	Obravnava v študiji
Raziskovalci tablo Kanban enačijo s tablo Scrum	
Tabla Kanban je nastala kot nadgradnja table Scrum	
Tabla Kanban se obravnava kot element Kanban metode	

Drugo:

4. Ali se je stanje obravnave Kanbana pristopa v literaturi po letu 2012 izboljšalo v primerjavi z letom 2012?

4.1. Prepreke uspešne implementacije Kanbana

Prepreka	Obravnavana v študiji	Opombe
Neskladje definicij		
Nejasne smernice		
Protislovja in neujemanja		

Druge prepreke, obravnavane v študiji:

Neskladje definicij: za elemente ne obstaja standardna definicija v razpoložljivi literaturi

Nejasne smernice: slabo določena navodila in smernice za oblikovanje in implementacijo elementov

Protislovja in neujemanja: elementi so v protislovju, oz. se njihova definicija ne ujema z drugim elementom ali načelom vitkosti.